

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
3 October 2002 (03.10.2002)

PCT

(10) International Publication Number
WO 02/078258 A2

- (51) International Patent Classification⁷: **H04L 12/18**, 29/06
- (21) International Application Number: PCT/US02/07200
- (22) International Filing Date: 8 March 2002 (08.03.2002)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
60/274,398 8 March 2001 (08.03.2001) US
- (71) Applicant (*for all designated States except US*): **DESK-TOP TV, INC.** [US/US]; 130 West Union Street, Pasadena, CA 91103 (US).
- (72) Inventors; and
- (75) Inventors/Applicants (*for US only*): **AHN, Myron** [US/US]; 50-B Peninsula Center Drive #329, Rolling Hills Estates, CA 90274 (US). **MORISON, Rodney** [US/US]; 1272 E. Calaveras St., Altadena, CA 91001 (US).
- (74) Agent: **NATAUPSKY, Steven, J.**; **KNOBBE, MARTENS, OLSON AND BEAR, LLP**, 620 Newport Center Drive, 16th Floor, Newport Beach, CA 92660 (US).
- (81) Designated States (*national*): AE, AG, AL, AM, AT, AT (utility model), AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, CZ (utility model), DE, DE (utility model), DK, DK (utility model), DM, DZ, EC, EE, EE (utility model), ES, FI, FI (utility model), GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SD, SE, SG, SI, SK, SK (utility model), SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZM, ZW.
- (84) Designated States (*regional*): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).
- Published:**
— *without international search report and to be republished upon receipt of that report*
- For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.*

(54) Title: EFFICIENT LARGE SCALE BROADCASTING ON LOCAL AREA NETWORKS SERVED BY HIGH SPEED TERRESTRIAL OR SATELLITE COMMUNICATION LINKS

(57) Abstract: An enhanced streaming content distribution system for tunneling multiplexed broadcastable data streams in a manner which reduces bandwidth demands imposed on a content server and a client network. The content distribution system may be used to distribute broadcastable streaming content across one or more subnets in a manner which does not require network components to be configured for multicast transmission. The content distribution system further provides data stream balancing among a plurality of clients through performance evaluation. Additionally, the system provides a flexible method for reducing streaming content interruptions using a combination of buffering and data stream splicing routines.



WO 02/078258 A2

DESKTV.002VPC

EFFICIENT LARGE SCALE BROADCASTING ON LOCAL AREA NETWORKS
SERVED BY HIGH SPEED TERRESTRIAL OR
5 SATELLITE COMMUNICATION LINKS

Background of the Invention

Field of the Invention

- 10 The present invention generally relates to network communications, and more particularly to a system and methods for efficient streaming of multimedia data to a plurality of users on a local area network using high-speed networking access and/or a satellite communications data downlink.

15 Description of Related Art

- As high-speed network services achieve widespread availability, the demand for new and more efficient methods of multimedia distribution continues to increase steadily. Currently, there is a significant interest in streaming multimedia content, including television and radio programming, across network connections as an alternative to more traditional broadcast methods. Network streaming of multimedia content has the potential to increase the accessibility of this information to users who could not otherwise receive it by other means, as well as, provide a convenient alternative for viewing or listening on personal computers without requiring separate devices such as television sets, cable or satellite decoders/set-top boxes, or radio receivers.
- 20
25

- Conventional transmission methods, including cable and satellite services suffer from a number of drawbacks which limit their ability to reach many users. In the case of cable services, potential subscribers may be unserviceable due to the expenses associated with laying cable and maintaining the necessary communications infrastructure. This problem is especially prevalent in business and commercial districts where creating and maintaining a cable network is prohibitively expensive. Furthermore, satellite services may also pose technical hurdles which are not easily overcome. For example, commercial satellite users
- 30

often must acquire roof rights and may not be positioned in a location capable of receiving the satellite signal.

The advent of high speed networking technologies such as broadband and T1/T3 lines, which permit access to Internet services including email and the World Wide Web (WWW) provide an infrastructure which may be configured to support streaming television and radio content as an alternative to conventional broadcasting methods. While broadband streaming has been developed to a certain extent, current implementations still suffer from several significant drawbacks which limit its utility and prevent widespread acceptance.

Typically, television and radio streaming consumes very large amounts of bandwidth and provides only marginal quality. Current implementations which provide broadcast quality streaming services place unreasonable demands on both public (Internet) and private (Local Area Network) network resources. Additionally, due to network latency, bandwidth saturation, and hardware failures (partial or whole) at network nodes or interconnects, a typical streaming broadcast often appears "choppy" and fragmented. Furthermore, it is currently not practical for many different broadcast channels to be sent across a network to permit viewing and rapid switching between channels in a manner similar to cable or satellite broadcasts. Thus, until the problems associated with broadcast over broadband networks can be resolved, there will likely be no solution for those consumers desiring an alternative to conventional transmission, cable, or satellite services.

A particular problem present in conventional network streaming results from the manner in which the signal is transmitted from a broadcast source to client stations. Generally, for each client station receiving the broadcast, an independent data stream must be maintained with the broadcast source. This topology rapidly increases the bandwidth requirements of the server as more clients are connected to the source for the purposes of receiving the broadcast signal. For example, a single network may contain many hundreds of clients each requesting a dedicated stream from the broadcast source. As a result, bandwidth demand increases with each client request resulting in a reduction in the quality of service to all clients.

In conventional systems, multimedia content distribution is often performed using a dedicated server computer to distribute streaming content to other computers within a local area network (LAN). This approach suffers from the

drawback that it may be undesirable or impractical to maintain a dedicated server computer and performance inefficiencies may arise as the number of client connections to the server increase. In many streaming configurations, the bandwidth and computational demands placed upon the server from multiple connecting clients exceeds that server's capability to provide streaming content of acceptable quality. Degradation of streaming performance is often observed as pauses or drop-offs which cause undesirable interruptions or latency in the data stream.

Another problem with this approach is that when the server goes down (due to various problems with source or intermediate network devices including bandwidth saturation, device failure, power interruption or maintenance requirements), the data broadcast to each client may be undesirably interrupted. Conventional streaming applications do not provide mechanisms for seamless switching to alternative content sources to maintain data stream integrity and thus each client receiving the streaming information may experience undesirable content interruption when the client switches sources.

Multimedia streaming across public networks, including the Internet, suffers from still further drawbacks which reduce the practicality of serving large numbers of remotely located clients. A principle problem lies in the manner in which the data stream must be packaged to distribute the information from a content server to one or more client computers requesting access to the multimedia content. The requirement for discrete data streams to be used in multimedia streaming applications places a significant demand on the network backbone, public infrastructure, and last-mile Internet gateway used to carry the information between the server and the clients. These bandwidth limitations diminish the ability to stream high-quality, multi-channel video and audio content to large numbers of users without compromising the performance of networks which carry the information.

In large networks having many client computers requesting streaming content, the problem of maintaining acceptable streaming quality is particularly apparent even when multiple servers are used. This problem is further complicated when multiple data streams corresponding to different channels are to be made available to the clients. In such an instance, seamless switching between

channels is often problematic and further results in undesirable delays and performance degradation.

Conventional methods for distribution of high-quality multimedia content also do not effectively deal with the problems of server overload and client balancing. These methods often rely on static network topologies that cannot adequately react to changing network conditions or computer loads. In a static network environment, a content server designated to stream data to requesting clients may become overloaded while other computers capable of performing server operations within the network remain substantially underutilized. Dynamic load balancing in a network such that the task of content streaming could be distributed over computers based on available bandwidth and computing capacity would be useful in reducing the bottlenecks and problems encountered with the static network design.

Recently, multicast methods for distributing high-bandwidth video and audio content have been described. In a multicast environment, a single data stream may serve more than one client simultaneously to thereby reduce the amount of bandwidth consumed. While multicast techniques may be applied in a local area network environment using dedicated routers and hubs, this manner of content distribution is generally not available over public networks, such as the Internet, because the underlying hardware which permits communication between computers is not practical to configure for this transmission mode. Consequently, conventional multimedia streaming is still limited when the content server resides outside a multicast-enabled network resulting in bandwidth imposed limitations that reduce the potential for reliable, high quality access to video and audio content.

25

Summary of the Invention

In one aspect, the invention comprises a method for encoding multimedia information for distribution across a communications network to a plurality of client devices wherein the method comprises the acts of (a) receiving the multimedia information in the form of at least one data stream from a stream source, (b) performing an operation to transform the data stream into a broadcastable data stream, (c) combining the at least one broadcastable data stream into a multiplexed data stream, and

30

(d) encoding the multiplexed data stream in a tunnel wrapper to generate an encoded multiplexed data stream that can be transparently transmitted across a network.

5 In another aspect, the invention comprises a method for distributing streaming content across a communications network to a plurality of client devices wherein the method comprise the acts of (a) receiving the streaming content in the form of a plurality of raw data streams from at least one stream source, (b) converting the plurality of raw data streams into a multiplexed data stream, (c) encoding the multiplexed data stream in a tunnel wrapper to generate an encoded
10 multiplexed data stream that can be transparently transmitted across a network, (d) transmitting the multiplexed data stream to at least one client, (e) receiving the multiplexed data stream at the client and subsequently decoding the encoded multiplexed data stream to generate the multiplexed data stream, and (f) broadcasting the streaming content corresponding to one or more of the raw data
15 streams contained in the multiplexed data stream using independent broadcast channels.

In still another aspect, the invention comprises a method for distributing streaming content across a non-multicast enabled communications network to a plurality of client devices wherein the method further comprises the acts of (a)
20 receiving the streaming content in the form of at least one raw data stream from a stream source, (b) converting the at least one raw data stream into a presentation compatible data stream, (c) performing a local multicasting operation to transform the formatted data stream into a broadcastable data stream, (d) combining the at least one broadcastable data stream into a multiplexed data stream, (e) encoding
25 the multiplexed data stream in a tunnel wrapper to generate a encoded multiplexed data stream that is transmitted across the non-multicast network, and (f) receiving and decoding the transmitted encoded multiplexed data stream by at least one client device which acts as a leader to further distribute the streaming content contained in the multiplexed data stream to the client devices in a broadcast mode.

30 In yet another aspect, the invention comprises a method for streaming multimedia data generated by a plurality of content sources over a communications network wherein the method further comprises: Transforming the multimedia data generated by each content source into a plurality of formatted datastreams,

converting the plurality of formatted data streams into a plurality of broadcastable data streams, packaging the plurality of broadcastable datastreams into a multiplexed data stream, and tunneling the multiplexed data stream through a communications network to at least one client device wherein the client device
5 subsequently recognizes the plurality of broadcastable datastreams and transmits the multimedia data contained therein to at least one client device in a broadcast mode.

In a further embodiment, the invention comprises a method for streaming data comprising a plurality of channels comprising multimedia content from a server to a plurality of clients wherein the method comprising: Transforming the
10 streaming data for each of the plurality of channels into broadcastable streaming data types at the server and further combining the broadcastable streaming data types into an encoded multiplexed datastream, transmitting the encoded multiplexed datastream across a communications network to a least one of the
15 plurality of clients which acts as a site leader, and decoding the multiplexed datastream by the site leader and thereafter re-transmitting at least one of the plurality of channels of streaming data in a broadcast mode to clients residing in the same subnet as the site leader.

In a still further embodiment, the invention comprises a method distributing
20 streaming content from a server to a plurality of clients in such a manner to preserve available bandwidth wherein the method further comprises: Designating at least one of the plurality of clients to act as a site leader wherein the site leader communicates with a remotely located server through a communications medium; Transmitting the streaming content from the server to the site leader; configuring
25 the site leader to receive the streaming content and re-transmit at least a portion of the streaming content over a first subnetwork in a broadcast mode, the site leader client further configured to transmit the streaming content to a subnet leader client associated with a second subnetwork in a guaranteed delivery mode wherein the subnet leader client re-transmits at least a portion of the streaming content over
30 the second subnetwork in a broadcast mode; Configuring a first plurality of clients residing within the first subnetwork to acquire the streaming content re-transmitted over the first subnetwork by the site leader wherein the broadcast mode provides shared accessibly to the streaming content; and Configuring a second plurality of

clients residing within the second subnetwork to acquire the streaming content re-transmitted over the second subnetwork wherein the broadcast mode provides shared access to the streaming content.

5 In yet another embodiment the invention comprises a method of distributing streaming content within a client network that comprises a site leader client and at least one additional clients and wherein the streaming content originates from a remotely located stream server wherein the method further comprises: Transmitting the streaming content from the remotely located stream server to the site leader; and Receiving the streaming content by the site leader and
10 subsequently replicating the streaming content for distribution to at least one client located in a local subnetwork using a broadcast transmission mode and further replicating the streaming content for distribution to at least one client located in a remote subnetwork using a guaranteed delivery mode.

In another aspect, the invention comprises a method for distributing an
15 incoming data stream within a client network wherein the incoming data stream is received from a server outside of the client network and wherein the client network comprises a plurality of clients distributed across at least one subnetwork, the method further comprising: Identifying one or more clients that have requested access to the incoming data stream;
20 Analyzing the clients on the basis of a performance metric wherein the performance metric is indicative of each client's ability to receive and distribute the incoming data stream; Forming a distribution arrangement based on the performance metric wherein the client with the highest performance metric is designated as a site leader and wherein at least one client with a lesser
25 performance metric located outside of the subnetwork where the site leader is located is designated as a subnet leader which establishes a guaranteed delivery connection to the site leader; and Configuring the site leader to receive the incoming data stream wherein the site leader replicates the contents of the incoming data stream and broadcasts the data stream contents to clients located in
30 the same subnetwork as the site leader and furthermore the site leader replicates the contents of the incoming data stream and transmits the replicated data stream to the subnet leader through the guaranteed delivery connection.

In a further aspect, the invention comprises a system for replicating and broadcasting a data stream in a client network wherein streaming content is transmitted from a server and is received by a client network, the system further comprising:

- 5 One or more clients that are interconnected within the client network wherein each client maintains a performance score that is indicative of the clients ability to replicate and distribute the data stream and wherein the client with the highest performance score is designated as a site leader to receive the data stream from the server and wherein the site leader is further configured replicate and re-
10 transmit the data stream to clients to thereby reduce the server bandwidth required to distribute the data stream to each client in the client network.

- In a still further aspect, the invention comprises a method for moderating streaming data between a content server and a plurality of clients so as to reduce data interruptions the method further comprising: Designating a first leader from
15 the plurality of clients; Transmitting the streaming data from the content server to the first leader wherein the first leader buffers at least a portion of the streaming data and subsequently distributes the streaming data to the plurality of clients; Designating a second leader from the plurality of clients upon receiving notification from the first leader of an impending streaming content distribution interruption
20 whereupon the content server performs a transmission switch to begin transmitting the streaming data to the second leader which buffers at least a portion of the streaming data; and Performing a distribution switch whereupon the first leader coordinates distribution of the streaming data with the second leader and wherein the buffered streaming data on each leader is used in conjunction with a splicing
25 operation to transparently switch between streaming data from the first leader to streaming data from the second leader.

- In yet another aspect, the invention comprises a method for moderating streaming content between a client acting as a content distribution source which receives streaming content from a content source and thereafter distributes the
30 streaming content to a plurality of clients so as to reduce data interruptions; the method further comprising: Designating a first subnet leader from at least a portion of the plurality of clients located in a different subnet from that occupied by the content distribution source; Transmitting the streaming content from the content

distribution source to the first subnet leader wherein the first subnet leader buffers at least a portion of the streaming data and subsequently distributes the streaming content to the plurality of clients; Designating a second subnet leader from the plurality of clients located in substantially the same subnet as the first subnet leader, which upon receiving notification from the first subnet leader of an impending streaming content distribution interruption, the content distribution source performs a transmission switch to begin transmitting the streaming content to the second subnet leader which buffers at least a portion of the streaming content; and Performing a distribution switch whereupon the first subnet leader coordinates distribution of the streaming content with the second subnet leader and wherein the buffered streaming content on each subnet leader is used in conjunction with a splicing operation to transparently switch between streaming content from the first subnet leader to streaming data from the second subnet leader.

In another embodiment, the invention comprises a system for maintaining substantially uninterrupted streaming content, the system further comprising: A streaming content distribution server which transmits streaming content; A first content leader which receives the transmitted streaming content from the streaming content distribution server and buffers at least a portion of the streaming content in a first stream buffer and thereafter distributes the streaming content to at least one client device; and A second content leader which receives the transmitted streaming content from the streaming content distribution server and buffers at least a portion of the streaming content in a second stream buffer and wherein at least a portion of buffered streaming content that is commonly contained in both the first stream buffer and the second stream buffer is identified as a stream jump position where a streaming content transition takes place such that the first content leader ceases to distribute the streaming data to the at least one client device and the second content leader commences with distributing the streaming content to the at least one client device at the stream jump position to provide the at least one client device with substantially uninterrupted distribution of streaming content.

In still another embodiment, the invention comprises a method for distributing streaming content transmitted by a content provider to generate

substantially uninterrupted streaming content distribution, the method further comprising: Buffering at least a portion of a first stream of streaming content transmitted by the content provider in a first buffer and buffering at least a portion of a second stream of streaming content transmitted by the content provider in a second buffer wherein the first stream of streaming content is substantially identical to at least a portion of the second stream of streaming content; and Joining the streaming content from the first and second streams at a position where the streams overlap in the first and second buffer to generate a continuous data stream such that the content stream is distributed as a continuous and uninterrupted sequence.

In yet another embodiment the invention comprises a method of switching a source of streaming content from a first content provider to a second content provider, wherein the streaming content is buffered by a content recipient and wherein the streaming content is arranged in a selected sequence, the method further comprising: Determining that a first source of streaming content provided by the first content provider is interrupted such that it will no longer provide streaming content to the content recipient; Requesting a second source of streaming content from the second content provider, wherein at least a portion of the streaming content provided by the first and second content providers is substantially the same; Receiving the streaming content from the second source of streaming content into a buffer wherein the buffer size is selected to allow simultaneous buffering of at least a portion of the streaming content provided by the first content provider and at least a portion of the streaming content provided by the second content provider; and Removing overlapping portions of the streaming content provided by the first and second content providers and appending the streaming content from the second content provider to the streaming content from the first content provider to generate substantially continuous and uninterrupted sequence of streaming content.

In another aspect, the invention comprises a method of distributing streaming content from a content source to a plurality of networked clients wherein each of the plurality of clients includes stream buffering and broadcast capabilities, the method further comprising: Determining a performance capability for each of the plurality of clients; selecting a current leader from the plurality of clients based

upon the determined performance capabilities; Delivering the streaming content from the content source to the current selected leader; Broadcasting the streaming content from the current selected leader to the remaining plurality of clients so that the current selected leader and the plurality of clients receive the streaming
5 content.

In still another aspect, the invention comprises a method for distributing streaming content to a plurality of clients to maintain consistency in data throughput, the method further comprising the steps of: (a) conducting a performance evaluation for each of the plurality of clients to identify clients with
10 acceptable performance characteristics;
designating the client with the highest performance as a site leader; (b) transmitting the streaming content from a stream server to the site leader; (c) broadcasting the streaming content from the site leader to the remaining plurality of clients; (d) periodically re-evaluating performance for the plurality of clients to
15 identify changes in client performance characteristics; and repeating steps (a)-(e) to thereby maintain service quality in the distribution streaming content.

In a still further embodiment the invention comprises a method of determining a leader among one or more clients in a subnet, the method further comprising: Assuming a client is the subnet leader unless another client is better
20 qualified to be the leader; Assessing leadership qualities of the clients at selected intervals; and Providing a transition period that allows leadership to be transitioned from one client to another.

In another aspect the invention comprises a method of determining a leader in a subnet having a plurality of clients wherein the leader receives a data stream
25 from a source and distributes the data stream to clients, the method further comprising: Assuming each client to be the leader unless another client is better qualified to be the leader based upon a performance assessment to thereby establish a current leader that is the most qualified of the plurality of clients based upon the performance assessment; Announces leadership by the current leader
30 recognized by non-leader clients which listen to the current leader's announcement periodically performing a performance assessment for each of the non-leader clients and comparing the performance assessment to the announcing leader's performance assessment to determine whether at least one of the non-leader

clients is better qualified to assume leadership within the subnet; and Providing a transition state that allows the non-leader client with the highest performance assessment greater than the current leader to assume leadership of the subnet as a new leader wherein the current leader becomes a non-leader client.

5 In a still further aspect the invention comprises a method of determining a leader among one or more members of a subnet, the method further comprising: Allowing each member to be the leader unless another member is better qualified to be the leader; Assessing leadership qualities of the members at selected instances; and Providing a transition period that allows transition of leadership from
10 one member to another.

In another aspect, the invention comprises a system for determining a leader in a subnet wherein the subnet comprises one or more clients that are capable of acting as the subnet leader, the system further comprising: A process that runs on each of the clients wherein the process determines a leadership
15 quality for the client and wherein the process periodically compares the leadership quality of the client to that of other clients within the subnet such that the client with the best leadership quality assumes leadership.

In still further aspect, the invention comprises a method of distributing streaming content from a content source to a plurality of networked clients, wherein
20 the plurality of networked clients are logically divided into a plurality of subnets, the method further comprising: Determining a performance capability for each of the plurality of clients;

Selecting, based upon the performance capability of each of the plurality of clients, a subnet leader for each of the plurality of subnets; Selecting, based upon the
25 performance capability of each of the plurality of subnet leaders, a site leader; Transmitting streaming content from the content source to the selected site leader; Transmitting the streaming content from the selected site leader to the plurality of subnet leaders; and Broadcasting the streaming content from the selected site leader and the plurality of subnet leaders to the clients within the plurality of
30 subnets.

In another embodiment, the invention comprises a method for improving distribution of streaming content between a content server and a plurality of clients, the method further comprising the acts of : (a) Identifying a data stream distribution

capacity for each of the plurality of clients; (b) Identifying a performance score for each of the plurality of clients indicative of the relative efficiency with which each client can broadcast the streaming content to other clients; (c) Arranging the clients in a hierarchical subnet ordering wherein the client with the highest relative efficiency is designated as a primary site leader which broadcasts streaming content to one or more subordinate clients having lesser performance scores and wherein the number of subordinate clients does not exceed the data stream distribution capacity of the primary site leader; and If additional clients remain, proceeding to act (d) where act (d) comprises designating the subordinate client with the highest relative efficiency as a subordinate site leader and arranging the remaining clients in the hierarchical subnet ordering wherein the number of subordinate clients under the subordinate site leader does not exceed the data stream distribution capacity of the subordinate site leader; and if additional clients remain, repeating act (d) until all clients have been arranged in the hierarchical subnet ordering.

In a still further embodiment the invention comprises a method of distributing a data stream to one or more clients in a client network wherein the client network receives the data stream from a server, the method further comprising the acts of: (a) assessing distribution capabilities of the clients and assigning scores to the clients accordingly; (b) selecting a highest score client to be a site leader that receive the data stream from the server and distribute the data stream therefrom; (c) if necessary, selecting a first group of clients to establish client-to-client fan-out connections to the site leader wherein the number of clients in the first group is determined by the site leader's distribution capability; (d) if necessary, forming fan out distribution of the data stream from one or more clients of the first group to remaining clients; and (e) repeating acts (a) to (d) as needed.

In another aspect, the invention comprises a method of organizing a distribution of a data stream within a client network having a plurality subnets wherein each subnet is represented by a subnet leader and wherein the data stream is sent from a server to the client network, the method performed by a selected subnet leader further comprises: Requesting a list of possible connection sources from the server or a current site leader that represents the most capable subnet leader for receiving the data stream from the server and distributing the

data stream therefrom wherein the connection sources comprise subnet leaders that are either already receiving the data stream or requesting the data stream; Receiving the list of connection sources; Analyzing the connection sources wherein a score is assigned to each source on the list and wherein the score is
5 based on the ability of the source to receive and distribute the data stream; Selecting the highest score source to be a new site leader such that the new site leader now receives the data stream from the server; Selecting a first group of subnet leaders to receive replicated data stream from the new site leader via a fan-out connection wherein each subnet leader of the first group forms a client-to-client
10 connection to the site leader; Distributing the data stream from each of the subnet leader of the first group to its corresponding subnet members; and If necessary, further distributing the data stream from selected subnet leader of the first group to remaining subnet leaders.

In a still further aspect the invention comprises a system for transmitting
15 streaming content via a non-multicast supported public network, the system further comprising: A streaming content source that receives a plurality of streams of streaming content wherein the streaming content source multiplexes the plurality of streams of streaming content into a formatted data stream and wherein the streaming content source wraps the formatted data stream into a protocol
20 supported by the public network and induces transmission of the wrapped formatted data stream via the non-multicast supported public network; A recipient network that includes a plurality of client devices, wherein the recipient network demultiplexes and unwraps the formatted data stream, wherein the recipient network transmits the unwrapped data stream to at least one of the client devices
25 so as to broadcast the streaming content.

In another embodiment the invention comprises a system for transmitting streaming content via a non-multicast supported public network, the system further comprising: A streaming content source that transmits a plurality of streams of streaming content via a broadcast or multicast satellite network wherein the
30 streaming content source multiplexes the plurality of streams of streaming content into a formatted data stream and wherein the streaming content source wraps the formatted data stream into a protocol supported by the public network and induces transmission of the wrapped single formatted data stream via the non-multicast

supported public network; and A recipient network that includes a plurality of client devices, wherein the recipient network demultiplexes and unwraps the formatted data stream, wherein the recipient network transmits the unwrapped data stream to at least one of the client devices so as to broadcast the streaming content.

5

Brief Description of the Drawings

These and other aspects, advantages, and novel features of the invention will become apparent upon reading the following detailed description and upon reference to the accompanying drawings. In the drawings, similar elements have similar

10

reference numerals.

Figure 1 is a schematic representation of a tunneling network environment used for streaming content distribution.

Figure 2A is another schematic representation of the tunneling network environment and content sources.

15

Figure 2B is another schematic representation of the tunneling network environment which includes a satellite transmission network.

Figure 2C is a block diagram of one embodiment of a server-side process associated with distributing streaming content using the tunneling network environment.

20

Figure 2D is a block diagram of one embodiment of a client-side process associated with distributing streaming content using the tunneling network environment.

Figure 3A is a block diagram for a client network connected to a server used for streaming content distribution.

25

Figure 3B is a block diagram illustrating a client functionality for streaming content reception and utilization.

Figure 4 is a block diagram illustrating a subnet having a client leader and one or more subnet client members.

30

Figure 5A illustrates one embodiment for distributing streaming content in the client network.

Figure 5B illustrates several protocols that may be used in the implementation of streaming content.

Figure 6A illustrates one embodiment of broadcasting streaming content in the client network.

Figures 6B-E illustrate one embodiment of a sequence for broadcasting streaming content in a client network.

5 Figure 7A is a block diagram illustrating one embodiment of a subnet leader connection structure.

Figure 7B is a flowchart illustrating a process for subnet leader connection.

Figure 8A is a schematic representation of one embodiment of a fan-out architecture for streaming content.

10 Figure 8B is a schematic representation of a fan-out architecture where subnet rearrangement takes place.

Figures 9A-C are flowcharts illustrating leader arrangement processes.

Figure 10 illustrates a method of determining a leader within a subnet.

Figure 11A is a flowchart illustrating leader election within a subnet.

15 Figure 11B is a flowchart illustrating a streaming content transition process.

Figures 12A-B illustrate one embodiment of a stream splicing operation used to preserve the integrity of the streaming content.

Figure 13 illustrates one embodiment of a stream jumping sequence used to preserve the integrity of the streaming content.

20 Figures 14A-F illustrate an exemplary stream splicing operation using a client buffer.

Figures 15A-B illustrate flowcharts for stream splicing operations that are used to preserve the integrity of the streaming content.

25 Detailed Description of the Preferred Embodiment

Tunneling Multiple Broadcastable Media Streams from Server to Client

30 Broadband multimedia applications such as those used for live audio and video streaming are bandwidth intensive operations. Generally, the quality of the multimedia stream is directly related to the amount of information that must be transmitted from a hosting source or content server to client listeners where the information is received and processed. For example, a typical high quality video

stream may require sustained data transmission rates of approximately 100 kilobytes/sec – 500 kilobytes/sec. As the number of transmitted streams increase, so too does the bandwidth requirement necessary to provide this data. Furthermore, with an increasing number of clients requesting streaming data from the content server, the bandwidth demands upon the server and the associated network infrastructure increases proportionally.

Aspects of invention overcome conventional data streaming limitations by addressing the fundamental problem of bandwidth consumption in a streaming multimedia environment. In one aspect, the invention reduces bandwidth imposed limitations by reducing the number of active data streams required to simultaneously transmit multimedia content to large groups of users. Additionally, the invention incorporates features that allow distribution of multiple channels of streaming content to provide each user with the ability to rapidly and seamlessly “switch” between channels in a manner that reduces delays commonly associated with acquiring different data streams.

In another aspect, bandwidth demands associated with streaming multimedia content are reduced through the use of broadcast transmissions received from a hosting source which has previously converted the streaming content from a another data type into a broadcast data type. The broadcast information is transformed from another protocol, such as TCP, UDP, or multicast, into a format that does not require multicast-enabled hardware or networking components for playback. Bandwidth is then preserved by using a broadcast transmission mode that is reliably resident in the networking components of the listening clients. In this architecture, a single data stream may serve one or more clients simultaneously without the requirement of establishing individual connections and transmitting multiple data streams between the clients and the content server.

In conventional multicast networks, the content server and clients are necessarily interconnected by a multicast compatible network configuration in order to distribute streaming content in a multicast mode. This requires the use of multicast-enabled routers and switches throughout the network that must be properly configured and maintained. Conventional multicast streaming is limited in many existing public and private network infrastructures, including Internet,

because these networks are not multicast enabled. Therefore, use of these networks precludes multicast streaming without significant modifications. In the case of the Internet, providing complete multicast retrofitting would be prohibitively expensive and time consuming.

5 Aspects of the invention overcome the aforementioned limitations through the use of a tunneling environment where streaming data is encoded in such a manner so as to allow it to be transmitted across a non-multicast network while retaining the desirable features of a broadcastable data type. Subsequently, the transmitted data is decoded and transformed into a format that allows the
10 streaming content to be distributed in both a broadcast manner on a local subnet as well as transmitted across one or more subnets using a non-broadcast IP protocol such as TCP. In one aspect, the multiplexed streaming data is received by a device capable of interpreting the data and distributing the information over a local subnet as discrete broadcast transmissions. The broadcast mode of
15 transmission is typically supported by most network configurations and retains the principle benefit of efficient data distribution to multiple devices using a single data stream. A further benefit to this approach is that streaming content may be encoded using conventional software applications which are limited to generating data in only unicast and multicast formats. Furthermore, the streaming content
20 may be multiplexed to encode multiple data channels or streams which are transformed into a single data stream. Tunneling of the multiplexed data stream allows the streaming content to be transparently transmitted across a non-multicast enabled network infrastructure and subsequently used to broadcast streaming content to a plurality of client computers maintained in a local broadcast-enabled
25 network. By reducing the number of data streams required to transmit multimedia content, the invention effectively reduces network congestion and increases the number of clients which can be simultaneously served by a content provider. Another benefit of the present system and methods described herein is that they provide a means for multiplexing a plurality of data channels or streams together
30 into a single composite data stream. The composite data stream desirably facilitates selective channel viewing as well as simultaneous presentation of the contents of two or more component data streams present in the multiplexed data stream.

Figure 1 illustrates an overview of a tunneling network environment 100. Typically, a content server 105 is responsible for collecting and distributing data streams to a plurality of client devices 110. The content provided by the server 105 may comprise a variety of different types of information including multimedia data, live video and audio broadcasts, pre-recorded video and audio, streaming data, and other information types that are desirably transmitted between the server 105 and the plurality of clients 110. In one aspect, this content is digitally encoded in a recognizable format and is interpreted by the client using an appropriate software application. Examples of some commonly recognized standards for multimedia encoding include MPEG (Moving Picture Experts Group), AVI (Audio Video Interleave), RM (Real Media), QT (Quick Time) and MP3 (MPEG Audio Level 3). It will be appreciated by one of skill in the art that the disclosed system and methods can be adapted for use with virtually any data format and is not necessarily limited to the aforementioned exemplary multimedia formats. Likewise, the type of data provided by the server are not necessarily limited to only multimedia data types such as audio and video data, but may also include other non-multimedia data types.

The content provided by the server 105 may further comprise more than one type or source of streaming content. For example, the content may comprise numerous audio or video feeds packaged together in a compound or multiplexed form. Compound data streaming is desirable in a number of applications including instances where a user may wish to view or listen to multiple feeds simultaneously. Additionally, compound data streaming can be used to provide a choice of selections to be determined by the user in a manner similar to changing channels on a television or switching between stations on a radio.

As previously described, the content provided by the server 105 is generally bandwidth intensive and is transmitted by the streaming of data to the clients 110. Data streaming is useful as it allows clients to view or listen to the incoming data stream without having to wait for an entire file to be downloaded. Additionally, data streaming is applicable to the reception of broadcast data that is not necessarily associated with a discrete file but rather transmitted continuously in a manner similar to a television or radio broadcast. The clients 110 generally receive the

data stream and, using an appropriate decoder, are able to present the data to the user in a recognizable form such as a video or audio feed.

In one aspect, multiplexed data streams 115 comprising one or more streams of encoded information are used to transmit streaming content to client networks 120 comprising discrete groupings of computers or content receiving devices remotely located from the content server 105. The use of multiplexed encoding desirably provides the ability to combine one or more channels or feeds of information into a single data stream and is used to distribute the content to one or more devices configured to receive the stream from the server 105. As will be described in greater detail herein below, transmission of the multiplexed data stream 115 is initiated by the client network 120 which establishes a connection with the server 105 through a site leader 125 and requests access to the streaming content. The site leader 125 is representative of an elected client 110 which is designated to provide the streaming content to the rest of the client network 120. Additional details of the mechanism of site leader election and function will be described in detail with reference to Figures 9-11.

Upon negotiation of a communication channel between the server 105 and the site leader 125, the server 105 commences with the transmission of the multiplexed data stream 115 to the site leader 125. Generally, the server 105 and the site leader 125 are remotely located from one another and are connected by way of a public network infrastructure 130, such as the Internet, which comprises hardware devices 135 including routers/switches which direct the multiplexed data stream 115 through a communications path that leads to the client network 120. The multiplexed data stream 115 transmitted by the server 105 through the public network infrastructure 130 is desirably encoded in a form so as to be recognizable and compatible with the hardware devices 135 used to implement the network infrastructure. A commonly implemented network connectivity protocol used for streaming content distribution between the content server 105 and the site leader 125 comprises the Transmission Control Protocol/Internet Protocol (TCP/IP). As will be described in greater detail hereinbelow, this protocol possess characteristics and features that may be utilized to transmit the single data stream in a guaranteed manner and provides mechanisms for error detection and retransmittal. Once the encoded multiplexed data stream 115 is received by the

site leader 125, the information contained in the stream is decoded and re-transmitted to the plurality of clients 110 using a broadcast compatible protocol such as the User Datagram Protocol (UDP).

In one aspect, the site leader 125 is configured to recognize the multiplexed data stream 115 and transform each channel contained in the multiplexed data stream 115 into a singular broadcast-enabled data stream 117. Each transformed data stream 117 may then be broadcast by the site leader 125 to the clients 110 which reside on the same subnet as the site leader 125. Broadcast streaming in this manner desirably makes use of the multiplexed data stream 115 in such a manner so that it may serve as a source of streaming content for one or more client devices which are not configured for multicast reception. Furthermore, by encoding the multiplexed data stream 115 in an appropriate format, streaming content transmission between the content server 115 and the site leader 125 can take place without the requirement for a multicast-enabled network infrastructure while the desirable features of packaging and broadcasting of multiple channels in a single data stream can be maintained.

Figure 2A illustrates an exemplary mechanism by which streaming content is acquired and distributed throughout the tunneling network environment 100. In one aspect, a plurality of content sources 205 provide information that is to be included in the available streaming content. The content sources 205 may include by way of example: broadcast feeds comprising audio and video content provided by satellite, cable, or transmission tower sources; multimedia streams from devices such as cameras, video cassette recorders/players, audio recorders/players; and other stream sources such as digitized audio and video files (MPEG, AVI, MP3, etc.) that may be stored both locally and remotely on computers or other devices.

Each content source 205 provides one or more raw streams 210 of information which may be distributed in a format native to the content source 205 or alternatively in another format compatible with other devices for which they are normally intended. The raw stream 210 further comprises information transmitted in a digitally encoded format which may, for example, comprise a video signal arising from a cable or broadcast television source. Likewise, the raw stream 210 may comprise a digital audio or video feed from a satellite source such as a commercial entertainment service provider. It will be appreciated that numerous

devices may generate various types and formats of raw streaming information that may be desirably collected for distribution to one or more clients in the form of streaming content.

5 A server-side data center 212 comprises hardware and software functionality necessary to convert the raw streams 210 into encoded multiplexed streaming content to be distributed to the client network 120. In one aspect, the data center 212 further comprises a stream encoder 215, a stream server 217, and a tunnel server 230 configured to exchange data and information with one another to generate and package the desired streaming content. It will be appreciated that
10 while each of these devices 215, 217, 230 is illustrated independently within the data center 212, the form and functionality of these devices may be readily combined in various manners. For example, the stream encoder 215 and stream server 217 may be integrated into a single hardware device or computer that performs the desired tasks associated with each device.

15 The stream encoder 215 receives the raw streams 210 and generates content data streams 220 to transform the information in the raw streams 210 into a format which is recognizable by other devices within the tunneling network environment 100. In one aspect, stream encoding may be necessary to convert streaming content from one data type into another such that the streaming content
20 can be presented to a user through a software application executed locally on each client. For example, the encoder 215 may function to convert both live and/or prerecorded audio, video, and data into a media format suitable for viewing or listening on the client devices. In one aspect, a commercially available encoder such as Microsoft® Windows Media™ Encoder (Microsoft, Inc., Redmond, WA),
25 Xing Mpeg Encoder (Real Networks, Inc., Redmond, WA), Quick Time Encoder (Apple, Inc. Cupertino, CA) may be used to convert the raw streams 210 into a compatible streaming media format such as Windows Media™ Format, MPEG, QuickTime®, Real Media® or other suitable media format. Typically, the content data streams 220 generated by the encoder 215 are encoded as discrete IP data
30 streams containing a singular source of content (i.e. one audio or video channel for each content data stream 220). As will be described in greater detail hereinbelow, these streams 220 are subsequently processed to transform them into a broadcast

compatible multiplexed stream which can be efficiently distributed to each client requesting the streaming content.

One significant function of the stream encoder 215 may be to perform various stream processing functions in which the raw streams 210 are modified to conform to a desirable format. For example, the stream encoder 215 may receive a raw stream 210 corresponding to a video signal having a preset size or color mapping. The stream encoder 215 may then re-size the resultant video image to achieve a new smaller image size (image reduction) or alter the audio or video quality. In one aspect, image reduction is useful for reducing the amount of information which is passed during content streaming while preserving acceptable quality audio and video quality. Furthermore, the stream encoder 215 may apply known compression techniques to modify the raw streams 210 and convert them to compressed formats thus reducing the amount of bandwidth required to transmit the streaming content. Additionally, the stream encoder 215 may convert the raw streams 210 from one format type to another which may be useful to provide a client-compatible stream or file type (i.e. conversion from MPEG to AVI, AVI to QuickTime, etc).

The one or more IP data streams 220 processed by the stream encoder 215 are subsequently transmitted to the media server 217, which performs additional stream processing routines related to data conversion to another protocol such as TCP, UDP, or multicast. In one aspect, the media server 217 receives IP datastreams from one or more stream encoders 215 and converts the IP data stream 220 into a broadcast-compatible data type. Each data stream 221 is subsequently transmitted to the tunnel server 240 using a protocol which allows the tunnel server 240 to convert the data into a broadcast compatible format. In one aspect, conversion of IP datastreams 220 may be performed by conventional applications designed to distribute streaming content in various transmission protocols, exemplified by TCP, UDP, and multicast. Exemplary applications which may be configured to receive input IP datastreams 220 and generate broadcast-compatible data streams 221 containing the streaming content in the desired form include Microsoft Media Server®, QuickTime Media Server®, Real Networks Media Server®, and other similar applications.

Generally, the broadcast-compatible data streams 221 are transmitted to the tunnel server 240 as discrete streams of information which are desirably packaged by the tunnel server 240 to generate a multiplexed stream comprising the contents of one or more broadcast-compatible data streams 221. The multiplexed stream information is thereafter encoded in a format which enables the multiplexed stream information to be transparently sent across conventional networks in such a manner so as to preserve a broadcast data format for the streaming content while "masking" this format from the network over which it is transmitted to insure compatibility. One desirable feature of this manner of streaming content transmission is that the use of a single encoded multiplexed stream 250 allows the client network 120 to simultaneously receive data corresponding to more than one video or audio signal through a single transmitted source. This feature further allows each client 110 to "switch" between channels present in the multiplexed stream 250 without having to acquire new streams of information. It will be appreciated that the use of the multiplexed stream format improves the flexibility of distribution of multimedia content and may be used, for example, to provide clients 110 with access to different channels or content through the use of a single data stream. Furthermore, the use of the multiplexed stream format provides a means for each client 110 to access multiple channels simultaneously. For example, this feature can be implemented to allow a user to view multiple video feeds simultaneously (i.e. two or more channels at once and picture-in-picture viewing) as well as sample discrete audio and video streams together (i.e. listen to an audio stream, while watching another separate video stream).

Typical connection-oriented communication requires that data be exchanged directly between systems resulting in increased bandwidth requirements to distribute streaming content to more than one client. For example, in order for four separate clients to receive and view a video stream using conventional connection-oriented communication methodologies, a separate data channel must be established between the server and each client. Thus, for a 100 Kilobyte/sec stream, approximately 400 Kilobyte/sec of bandwidth is required to service the 4 clients. As the number of clients increase, the bandwidth of the network to which the clients are connected is rapidly saturated which may lead to

decreased streaming performance across all clients as well as preventing additional clients from receiving the streaming content.

The method of stream broadcasting makes use of the encoded multiplexed data stream transmissions provided by the tunnel server 240 in a manner which does not incur the aforementioned incremental bandwidth penalties. In one aspect, the multiplexed data stream 250 comprises a broadcast-enabled format derived from a data type which is subsequently interpreted to efficiently transmit streaming content. Broadcast transmission of streaming information may further be used to serve content to a plurality of clients configured to receive the broadcast transmission while consuming only the relative bandwidth necessary to distribute streaming content to a single client. In one aspect, broadcast transmissions are implemented in the streaming environment where a single broadcast data stream can be "shared" among each client having an appropriate configuration. Thus, for networks having many clients each requesting access to similar streaming content, relatively little additional bandwidth is required to distribute the streaming content to all of the clients within the network. Aspects of the invention make use of this broadcasting functionality to distribute streaming content more efficiently than conventional connection-oriented streaming environments and is not limited for use in multicast enabled networks. A principle feature of the invention relates to the ability to combine both connection-oriented content streaming with broadcast-enabled content distribution to provide "multicast-like" functionality in a network that has not been implemented to support multicast transmissions.

In one aspect, problems associated with distributing streaming content over a network such as the Internet are overcome through the use of a network tunneling implementation. Network tunneling is used to create a virtual network connection between two or more computers wherein encoded data transmissions and information are transparently carried over a non-encoded network. Encoding data in this manner preserves the underlying content and format of the encoded information which may be subsequently decoded and recovered. Using the network tunneling protocol to encode streaming content provides a number of desirable features which add to the flexibility and utility of the invention. One feature of network tunneling is that it creates a secure transmission means wherein streaming information is protected from decoding or viewing by those for which the

data was not intended. Furthermore, the tunneling protocol is able to effectively “mask” underlying multiplexed and broadcast streaming content to enable it to be transmitted over conventional networks and devices without the need to implement specialized hardware or software functionality. The use of the tunneling protocol can also be implemented to conveniently allow passage of data through firewalls and other security measures which may be in place within a network without the need for substantial reconfiguration or administration of the network. Additionally, the tunneling protocol can be transmitted using a commonly recognized protocol such as Hypertext Transfer Protocol (HTTP). HTTP is a desirable protocol to be used for this purpose because it is supported by most existing networks and is considered a “normal” network traffic type which most computers and devices are compatible with.

In one aspect, network tunneling of the streaming content is accomplished using the tunnel server 240. The tunnel server 240 receives streaming data from either the stream encoder 215 or the media server 217 representative of streaming content which is to be distributed to one or more client networks 120. As previously indicated, the tunnel server 240 prepares a multiplexed data stream using one or more multicast data streams 221 transmitted by the media servers 217. The tunnel server 240 is further responsible for performing a tunnel encoding operation that is used to transform the multiplexed data stream into a format compatible with non-multicast enabled networks 130. As previously indicated, the encoded streaming content desirably retains its multiplexed properties which are made transparent to the network 130 through which the encoded data will travel and can be subsequently broadcast within a local subnet as one or more discrete channels of streaming information.

In one aspect, the tunnel server 240 encodes multiplexed data representative of the streaming content to be distributed to the one or more client networks 120 in a “wrapper” to provide compatibility with existing network infrastructures 130. The wrapper overlays and packetizes the multiplexed data and may include header/footer information which is recognizable by the network 130 over which the encoded multiplexed data 250 is sent. The use of a virtual network tunnel 260 therefore allows the server 240 to communicate with the client network 120 through other network connections, while encapsulating the streaming

content to preserve the multiplexed and broadcast properties of the data stream in a secure, flexible, and transparent manner. The encoded information is subsequently recovered in the client network 120 which in turn enables the multiplexed data transmissions to be separated and broadcast as discrete streams of content. Furthermore, the encoded multiplexed data 250 may be forwarded to other subnets for broadcast distribution thereby overcoming conventional subnet localized broadcast implementations. Additional details of the manner in which the multiplexed data is decoded and broadcast over each local subnet will be described in greater detail hereinbelow.

Figure 2B illustrates another embodiment of an exemplary mechanism by which streaming content is acquired and distributed throughout the tunneling network environment 101. Here the tunneling network environment incorporates a satellite distribution mechanism which performs one or more data stream distribution operations. In one aspect, the functionality of the tunneling network environment incorporating the satellite distribution mechanism possesses similar functionality as the tunneling network environment described in Figure 2A above. In this implementation, illustrated in Figure 2B, a satellite-based broadcast or multicast network 214 is logically interposed within the components of the data center 212. In one aspect, the satellite-based broadcast or multicast network 214 desirably is used to transmit data from the media server 217 to the tunnel server 242.

Because conventional satellite broadcast network topologies can be configured to support many different protocols including one-way IP protocols, such as UDP and multicast protocols, the satellite broadcast network topologies can be readily integrated into the communications network and therefore distribute broadcastable streaming content in an analogous manner to the conventional communications mediums described with reference to Figure 2A above. Therefore, as will be appreciated by one of skill in the art, the topological distinction between the satellite broadcast network and a conventional wire-based network is such that both may be used to support Internet Protocol level (IP-level) communications to distribute the broadcastable content between the devices of the data center 212. Furthermore, the satellite transport method may be configured to

provide a lower layer of transmission support which is transparent to IP communications.

In one aspect, formatted data or information (which may be representative of packetized information) is transmitted through an uplink dish to the satellite where it is subsequently transmitted to, and received by, a satellite decoder 242. The satellite decoder 242 comprises an integrated tunnel server functionality which may be used to encode the formatted data or information into a multiplexed data stream which is further encoded using a tunnel wrapper similar to that described above with reference to the tunnel server 240 (Figure 2A). In one aspect, the components necessary for performing these operations may include a desktop computer configured with a satellite receiver card. Alternatively, the satellite decoder/tunnel server 242 may be built using other commercially available components and devices that can be readily integrated into the enhanced streaming content distribution system and methods described herein.

Figure 2C illustrates one embodiment of a server-side process associated with distributing streaming content using the tunneling network environment 100. In one aspect, the data center 212 is configured to perform an application process that receives data 267 from one or more content sources 205. As previously indicated, the stream encoder 215 may receive this data, comprising raw streaming information, and perform one or more operations associated with transforming the raw streaming information into an application-compatible datastream. The application-compatible datastream may further comprise any of a number of different content formats including for example, MPEG, AVI, QT, or MP3 content formats. In one aspect, each datastream is associated with a discrete channel of information which may be further be packetized for network transmission. In the illustrated embodiment, two discrete datastreams comprising are shown which correspond to a first data channel 268 (Channel A) and a second data channel 270 (Channel B).

Subsequent to the datastream transformation, the data center 212 may perform a network process 269 to transfer the datastreams to the tunnel server. The transfer may be performed by a multicast transmission 271a, a UDP transmission 271b, or a TCP transmission 271c in manners described below.

Preferably, the two datastreams 268 and 270 remain as discrete datastreams during the transmission to the tunnel server.

As illustrated in Figure 2C, the multicast transmission 271a comprises manipulating the datastreams so as to convert its contents into a multicast format which may include addition of multicast header information 272a which is appended to the first and second data channels 268, 270. This operation 271a is desirably performed locally within the data center 212 to minimize the number of devices and network components that require multicast enablement. In one aspect, multicasting of the datastreams is conducted using one or more commercially available media servers which are configured to receive discrete datastreams and multicast the information contained therein to the tunnel server 240.

The datastreams may also be transmitted to the tunnel server via the UDP transmission 271b that comprises manipulating the datastreams so as to convert its contents into a UDP broadcast format which may include addition of UDP broadcast header information 272b which is appended to the first and second data channels 268, 270. Similar to the aforementioned multicasting mode of transmission, UDP broadcasting of the datastreams is conducted using one or more commercially available media servers which are configured to receive discrete datastreams and multicast the information contained therein to the tunnel server 240.

The data streams may also be transmitted to the tunnel server via the TCP transmission 271c that comprises manipulating the datastreams so as to convert its contents into a TCP format which may include addition of TCP header information 272c which is appended to the first and second data channels 268, 270. Similar to the aforementioned multicasting mode of transmission, TCP transmission of the datastreams is conducted using one or more commercially available media servers which are configured to receive discrete datastreams and multicast the information contained therein to the tunnel server 240.

Upon receiving the discrete datastreams via one of the aforementioned modes, the tunnel server in one aspect performs a broadcast translation 273, a multiplexing of streams 275, and tunnel encoding 278 in a manner described below. As illustrated in Figure 2C, the broadcast translation 273 comprises

manipulation of the datastreams so as to strip away the header information that was added in the network process 269 described above. In particular, the header (multicast header 272a, UDP broadcast header 272b, or TCP header 272c) is removed so as to yield the first and second data channels 268, 270 that are
5 substantially similar to that of the receive data 267 stage described above.

Following the broadcast translation 273, the datastreams are processed in the multiplexing operation 275 that transforms the discrete datastreams into a composite or multiplexed datastream 274. In one aspect, the multiplexed datastream 274 desirably consolidates the information for each of the discrete
10 datastreams into a singular datastream such that the two exemplary discrete datastreams 268, 270 may be combined and transmitted to the client network as a single stream of content. Multiplexing of the datastreams in this manner improves the ability to manage stream content distribution to client networks which may request or desire simultaneous access to more than one channel of information. In
15 one aspect, the multiplexed datastream comprises a combination of the discrete datastreams, but does not have any header information added to it.

Upon generating the multiplexed datastream 274, the tunneling operation 278 is performed to thereby encode the contents of the multiplexed datastream 274. In one aspect, encoding of the multiplexed datastream results in the
20 integration of a tunnel header 279 into the multiplexed datastream. The tunnel header 279 may provide routing or addressing information as well as other information required to encode the multiplexed datastream in a form which is transparent to the network which will be used for its transmission to the client network. Tunnel encoding desirably preserves the underlying streaming content
25 and "masks" this information from interpretation by devices and networking components aside from the client device for which the tunneled multiplexed datastream is intended. Tunneling of the multiplexed datastream in this manner can further be implemented using a commonly recognized transmission format such as TCP transmission using an HTTP protocol. The use of the commonly
30 recognized transmission format desirably insures that the tunneled information will be capable of being transmitted between a remotely located server and the receiving client. HTTP transmission of tunneled multiplexed data additionally facilitates transmitting the data across communications networks which may

implement firewalls or other security measures designed to block or filter other types of transmitted information. In one aspect, by encoding the underlying streaming content in a tunnel wrapper, one or more channels of broadcast-ready streaming content can be readily distributed to a client network while circumventing
5 conventional hardware limitations and security measures which may prohibit the transmission of unencoded multiplexed broadcast-ready streaming content.

Figure 2D illustrates a client side process 280 that receives and processes the tunnel-encoded multiplexed datastream that was formed in the manner described above in reference to Figure 2C. The client process 280 follows the
10 various operations that can be applied to the tunnel-encoded multiplexed datastream. In one aspect, the aforementioned operations may be performed by a site leader, a subnet leader, a client in a subnet, or other processes whose functions and characteristics are described in greater detail elsewhere in this paper.

As shown in Figure 2D, receiving and decoding operation 281 is first applied to the tunnel-encoded multiplexed datastream, wherein the tunnel header 279 is removed so as to yield a multiplexed datastream 293 that is substantially similar to the multiplexed datastream 274 (Figure 2C) that results from the multiplexing operation 275 performed by the tunnel server. Preferably, the receiving and
15 decoding operation 281 is performed by the site leader of the client network.

At this point, the multiplexed datastream 293 can be further distributed or be processed for presentation. If the multiplexed datastream 293 is to be distributed, it can be done so by either connection oriented guaranteed delivery or by broadcasting. In one aspect, the guaranteed delivery is preferably utilized to
25 distribute the datastream to subnet leader(s), and broadcasting is preferably utilized to distribute to clients within the same subnet as the member that is performing the operation 281. These two possible modes of distribution are depicted by a query state 282 that asks whether the multiplexed datastream 293 is to be directed to a subnet. If "yes", a guaranteed delivery processing operation
30 283 is performed on the multiplexed datastream 293, wherein a guaranteed delivery header 286 is added to the multiplexed datastream 293. In one aspect, the guaranteed delivery header 286 is a TCP header that is substantially similar to the TCP header 272c described above in reference to Figure 2C. As previously

described, the TCP type data is widely supported such that the guaranteed delivery using the TCP header can be realized without requiring undesirable reconfiguration of the client network. Following the guaranteed delivery processing operation 283, a transmit operation 285 transmits the multiplexed datastream with
5 the guaranteed delivery header 286.

If the multiplexed datastream 293 is not to be transmitted to a subnet, as indicated by "no" in the query state 282, a stream decoding operation 287 manipulates and demultiplexes the multiplexed datastream 293. As illustrated in Figure 2D, the operation 287 results in restoration of the discrete first and second
10 data channels 268 and 270. Thus in one aspect, the stream decoding operation 287 is a reverse process of the multiplexing operation 275 (Figure 2C) that takes place in the tunnel server.

Following the stream decoding operation 287, the demultiplexed datastream can either be broadcast with a subnet or used for local presentation. This decision
15 is depicted by a query state 288 that determines whether to broadcast. If the answer is "yes", a broadcast operation 288 adds the UDP broadcast header 290 to each of the discrete data channels in a manner that is substantially similar to the UDP transmission operation 271b of Figure 2C. The discrete datastreams are then broadcast within the subnet, and the receiving clients can "tune in" to a
20 selected channel as desired.

If the answer is "no" to the query 288, the discrete datastreams are not to be broadcast, and another query 291 is made wherein determination is made as to present the datastreams locally. If the answer is "yes", the discrete datastreams are presented locally in operation 292, wherein the discrete datastreams remain
25 substantially unaltered from the stream decoding operation 287. The local presentation operation 292 may involve, for example, a subnet leader simultaneously utilizing the datastreams and broadcasting it to the clients in the subnet. In one aspect, the subnet leader may also receive and utilize the datastreams not by direct local presentation, but rather by broadcasting and then
30 receiving its own broadcast just like other members of the subnet.

Efficient Client-side Replication, Distribution, and Broadcast

One aspect of the invention relates to a client network configured to efficiently replicate, distribute, and broadcast streaming data obtained from a remote server. Efficient distribution of streaming data permits one or more clients comprising the client network to be advantageously connected to the server by a single streaming connection capable of serving content to each of the clients. This feature of the invention is a notable improvement over conventional streaming technologies where each client must individually connect to the server thereby establishing numerous connections which increase the bandwidth load upon the server. Aspects of the invention utilize a single stream of content provided by the server to distribute streaming content in such a manner that the bandwidth consumed by providing a data stream to a single client is substantially the same as the bandwidth consumed when providing a data stream to multiple clients within the same local network.

In general, the below-mentioned components of a server, a site leader and client devices comprise computers configured to communicate through one or more different communications mediums. Each computer may further be represented by any microprocessor or processor controlled device that permits access to the communications medium and may include, by way of example, terminal devices; such as personal computers, workstations, servers, mini-computers, main-frame computers, laptop computers, a network of individual computers, mobile computers, palm top computers, hand held computers, a set top box for a TV, an interactive television, an interactive kiosk, a personal digital assistant, a communication device, an interactive wireless communications device, or a combination thereof. The computer may further comprise input devices such as a keyboard or a mouse, and output devices such as a computer screen or a speaker. Furthermore, the computer may serve as clients, servers, or a combination thereof.

Additionally, the computer may include an addressable storage medium or computer accessible medium, such as random access memory (RAM), an electronically erasable programmable read-only memory (EEPROM), random access memory (RAM), erasable programmable read-only memory (EPROM),

hard disks, floppy disks, laser disk players, digital video devices, compact disks, video tapes, audio tapes, magnetic recording tracks, electronic networks, and other devices to transmit or store data.

In one embodiment, the computer is equipped with a network communication device such a network interface card, a modem, or other network connection device suitable for connecting to the communications medium. Furthermore, the computer should run an appropriate operating system such as; Microsoft® Windows® 3.1, Microsoft® Windows® 98, Microsoft®, Windows® 98 Second Edition®, Microsoft® Windows® Millennium Edition®, Microsoft® Windows® NT, Microsoft® Windows® 2000, Microsoft® Windows® XP, Microsoft® Windows® CE, PalmOS®, Apple® MacOS®, Linux®, Solaris®, IRIX®, UNIX® or IBM® OS/2® operating system.

As is conventional, a preferred operating system further includes a TCP/IP stack or other network communications protocol which handles incoming and outgoing streaming content or data passed over the communication medium. In other embodiments, the operating system may differ depending on the type of computer, however, the operating system will continue to provide the appropriate network communications protocol necessary to establish communication links with the communications medium.

As previously described, the communication medium may include the Internet in the illustrated embodiment. The Internet is a global network of interconnected computers capable of exchanging information and data. The structure of the Internet, which is well known to those of ordinary skill in the art, includes a network backbone comprising communications channels such as copper wire or optical fiber based interconnections between numerous computers, hubs, and routers. The network backbone, and component devices therein, control, direct, and maintain information passed between computers. Additional networks may branch from the above-mentioned backbone with these branches, in turn, have sub-networks branching from them, and so on.

Typically, information is passed through the network in the form of packets which are discrete pieces the information desirably sent through the network. These packets comprise information encoded in a form interpretable by the network infrastructure and may support features such as data compression,

encryption and error correction to optimize the speed and efficiency by which the information is transferred. For a more detailed description of the structure and operation of the Internet, the reader is referred to "The Internet Complete Reference," by Harley Hahn and Rick Stout, published by McGraw-Hill, 1994.

5 One of ordinary skill in the art will recognize that the network of computers connected through the communications medium may be advantageously comprised of other types of networks without detracting from the invention. The communication medium and computer network may include, by way of example,
10 local area networks (LANs), wide area networks (WANs), public internets, private internets, a private computer network, a secure internet, a private network, a public network, a value-added network, interactive television networks, wireless data transmission networks, two-way cable networks, interactive kiosk networks, and the like. The disclosed invention is thus suitable for distributing streaming content through many different forms of communications mediums, however, the invention
15 use will be further disclosed in the context of distributing streaming content through the Internet.

Figure 3A illustrates a server-client network arrangement 300 comprising a client network 302 connected to a server 304 via a network connection 306 that allows streaming content to be transferred from the server 304 to the client network
20 302. The client network 302 comprises a plurality of clients 308 among which there is a lead client referred to as a site leader 310. In one aspect, the site leader 310 is selected from the available clients 308 within the client network 302 as being the most capable or available to establish a connection 306 with the server. The site leader acts as an intermediary between the server 304 and the client
25 network 302 wherein streaming content and information intended for the client network 302 is passed through the site leader 310 without individual clients connecting directly to the server 304. The site leader 310 is also responsible for requesting streaming content from the server 304, decoding streaming content transmitted by the server 304 and processing the streaming content for
30 presentation to a local user (if necessary). Additionally, the site leader 310 may distribute the streaming content throughout the client network 302 to enable other clients 308 to receive the desired information without the requirement of establishing new connections to the server 304. Thus, the network connection 306

depicted to connect the server 304 to the site leader 310 further serves as the source of streaming content for all of the clients 308 within the client network 302.

As previously described, information streamed by the server 304 may be desirably transmitted to the site leader 310 through a tunnel connection 260
5 wherein communications between the server 304 and the site leader 310 is encoded to preserve a network packet data type which may be used in streaming content distribution within the client network 302. The manner in which a site leader 310 is determined within the client network 302 is described elsewhere in the section entitled "Methods of Optimized Leader Election and Leader Feasibility
10 Evaluation for Multiple Clients".

In a typical client network 302, connections to the remote server 304 as well as connections within the network are routed through various network components which may include routers, switches, hubs, cables, optical lines, and/or other network components including wireless devices and access points which enable
15 network connectivity in a manner known in the art. For illustration simplicity, connections described throughout this document do not always show such devices but rather direct connections may instead be depicted to illustrate connectivity between components. For example, network connectivity between the server 304 and the site leader 310 is shown as the network connection 306. Furthermore,
20 other network connections may exist which interconnect each client within the client network 302. These network connections allow communication between the site leader and the one or more clients within the client network 302 and provide a means for distributing the streaming content transmitted by the remote server 304 and received by the site leader 310 which is thereafter distributed to those clients
25 308 which request access to the streaming content.

Figure 3B illustrates an exemplary client functionality used in the context of data stream distribution. In one aspect, the client 308 receives a data stream 312 from an upstream source, and may present the content of the data stream to an end user. Additionally, the client 308 may re-distribute the content of the data
30 stream 314. The client 308 as described and illustrated in Figures 3A-B is a generic term that may represent functionally different types of members which may be interconnected to form the client network 302.

The client network 302 may be comprised of a single client 308 or may contain one or more additional clients 308. In one aspect, a client network 302 is a dynamic structure which may grow and contract as clients enter and leave the network. For example, a client network 310 may initially comprise a single client
5 308 configured to receive streaming content which is thereafter joined by additional clients 308 which request access to the streaming information thereby increasing the size and complexity of the client network 308. As illustrated in Figure 4A, the subnet 320 comprises a plurality of clients 308 each of which is a subnet member 322. A subnet leader 324 may further be designated as a client within the subnet
10 320 that receives streaming content from an upstream source and distributes the streaming content to other members of the subnet 320. The manner in which the subnet leader 324 is elected is described elsewhere in the section entitled "Methods of Optimized Leader Election and Leader Feasibility Evaluation".

Figures 5A-B illustrate one embodiment of a data stream distribution
15 method 330 and the corresponding data stream protocols 340 that may be used to broadcast streaming content within the subnet 320. In one aspect, streaming content is distributed to all clients within the subnet 320 by decoding the streaming content received from an upstream source (i.e. a remote server, site leader, or other subnet leader) which has been encoded in a broadcastable format. The site
20 leader 324 possesses the functionality for receiving the encoded data stream which in one embodiment is tunneled through the existing network infrastructure. Tunneling in this manner preserves the underlying streaming content and the format in which it encoded. In one aspect, the tunneled streaming content is encoded to preserve a broadcastable data type. Upon receiving the tunneled
25 streaming content, the site leader 324 decodes the information and thereafter broadcasts the streaming content based on the broadcastable data type and the presence of other clients requesting access to the streaming content.

Figure 5A illustrates exemplary data distribution capabilities for each client 308. As will be described in greater detail hereinbelow, each client contained in
30 the local network may desirably operate as a site leader or subnet leader whereby the client may perform operations related to streaming content distribution within the network in addition to operating as normal client which receives and presents data to a user. Each client 308 is therefore desirably configured to permit it to

maintain a connection 332 to one or more upstream sources when acting as a site leader or subnet leader. The connection 332 established by the site leader or subnet leader may further be characterized by one of several different types of guaranteed delivery connections or protocols. Using these protocols, consistency and accuracy of the streaming content and information is desirably maintained through conventional error checking and packet ordering functions. In one aspect, streaming content is received by the client acting as a site leader or subnet leader using TCP/IP communications. As will be appreciated by one of skill in the art, TCP/IP communications protocols provide a mechanism for guaranteed delivery of packetized information such as the aforementioned streaming content.

Upon receiving the streaming content from the upstream source, the client 308 may present the streaming content to the user and additionally distribute the streaming data to other clients. In one aspect, streaming content may be replicated by the client and re-directed as needed to distribute the information to other clients within the local network. When distributing the streaming content to other clients located outside the subnet in which the client resides a guaranteed delivery connection 334 is desirably established. As previously indicated use of guaranteed delivery protocols desirably preserve the stream integrity during replication so that it may be passed multiple times without encountering degradation of the information contained in the data stream. Typically, the guaranteed delivery connections utilized throughout the local network require discrete data streams to be established between two clients that are to exchange the streaming content. In one embodiment, it is desirable to minimize the number of connections of this type that are used in the local network in favor of broadcast connections when possible to distribute streaming content more efficiently and to reduce network congestion.

When desirable, the streaming content received by the client 308 may be replicated and broadcast 336 within the local subnet in which the client resides. Broadcast of streaming content in this manner is desirably accomplished using a broadcastable stream which has been previously encoded by the remote server and transmitted to the site leader through the tunnel connection. The broadcastable format of the streaming content may further be preserved and passed between leader clients such that upon receiving the broadcastable data, the client may broadcast this information throughout the subnet in which it resides

to allow the content may be received by any listening client(s). Broadcast distribution of broadcastable streaming content is depicted by the broadcast element 336. Broadcasting in this manner is desirable as it allows the streaming content to be distributed to one or more clients using substantially the same amount of bandwidth. Thus, a significant decrease in the amount of network traffic required to service each individual client is realized as well while maintaining a reduced number of connections to the remote server.

The advantages provided by both the guaranteed delivery protocol and the broadcast protocol are desirably combined to distribute a single data stream received from a remote server to a plurality of receiving clients in the local network. Additional details of the distribution arrangement for streaming content based on these protocols is described below in reference to Figure 6A.

Figure 5B illustrates data stream protocols 340 that may be implemented in the various connection types described above in reference to Figure 5A. In particular, the connection between the server 304 and the site leader 310 is typically a guaranteed delivery connection 342, wherein the data stream is transmitted using Hypertext Transfer Protocol (HTTP) over a TCP/IP communications link. HTTP distribution of the streaming content desirably facilitates transmissions through firewall protected networks and is universally recognized through public networks including the Internet. Thus, the streaming data may be advantageously transmitted between the remote server and a client through a firewall enabled network without substantial need for reconfiguration or usage of a special protocol. Furthermore, the use of the HTTP data type desirably supports the tunneled network connectivity described above with reference to Figures 1-2 to permit broadcastable data to be transmitted over a non-multicast enabled network such as the Internet.

Like the connection between the remote server and the site leader, the connection between the site leader 310 and a subnet leader client 324 is desirably implemented using a guaranteed delivery connection 344. As before the data stream may use TCP/IP communications in connection with an HTTP protocol to insure content stream integrity and to preserve broadcastable data types.

Finally, the distribution of streaming data from the subnet leader 324 to the end user subnet member or client 322 is typically performed by a broadcast

protocol 346 when possible to more efficiently distribute streaming content as compared to guaranteed delivery protocols. In one aspect, broadcast of the streaming content is accomplished using a broadcastable data stream which is transmitting using the User Datagram Protocol (UDP). As is known in the art UDP
5 broadcasting may be desirably configured to distribute broadcastable data, including certain multicast-compatible UDP multimedia formats to thereby distribute content throughout the clients of the subnet while consuming bandwidth substantially equivalent to a single data stream.

Figure 6A illustrates an exemplary distribution configuration 350 that may be
10 used to distribute streaming data throughout a client network through utilization of the distribution methods and data stream protocols described above in reference to Figures 3-5. Indicated next to various exemplary connections and broadcasts are exemplary bandwidth consumption statistics which may arise when distributing the streaming content. This bandwidth information demonstrates that a significant
15 reduction in overall bandwidth utilization may be realized when the streaming content is distributed via this combination of protocols and permits more clients to receive the data than would be allowed if only connection-oriented (TCP based) protocols were used.

Within the local network 302, an exemplary client bandwidth limit
20 corresponding to a maximum available bandwidth for each client of approximately 400 Kbytes/s (Kbps) is observed. The total incoming and outgoing traffic for any given client within the local network is configured so that this limit is desirably not exceeded and in most cases the amount of bandwidth consumed is substantially less than this amount. Aspects of the invention desirably feature the ability to
25 utilize a single data stream transmitted from the remote server which is replicated and rebroadcast throughout the local network to service the plurality of clients. Furthermore, streaming content distribution within the local network is desirably configured in such a manner so that the bandwidth limit of each client is not exceeded.

30 As previously described and illustrated in Figure 6A, the remote server 304 is connected to the site leader 310 by a networking connection 342 which may include the Internet. In one aspect, the network connection between the remote server 304 and the site leader 310 utilizes HTTP communications to transmit

streaming information which may further include broadcastable data which has been tunneled through the networking connection 342 to preserve the broadcastable qualities of the data stream. The site leader 310 receives the single data stream corresponding to the streaming content and distributes it such that
5 numerous downstream clients in the network 302 receive substantially the same data stream contents. To further exemplify the types of communications which may occur within the local network, an exemplary network 302 comprises the site leader 310, three terminal clients 364a-c, and first and second subnets 360, 362 (each having additional terminal clients 372a-d and 382a-c). In one aspect, the
10 terminal clients 364a-c, 372a-d, and 382a-c are end user clients which may receive the streaming content but need not re-transmit the data to other systems. Furthermore, the terminal clients 364a-c, 372a-d, and 382a-c may be configured to operate in a broadcast-receive mode where each terminal client is able to receive broadcastable information. The first subnet 360 comprises a subnet leader 370
15 and the four terminal clients 372a-d which receive streaming content in the form of broadcast transmissions from the leader 370. Likewise, the second subnet 362 comprises a subnet leader 380 and three terminal clients 382a-c which receive broadcast streaming content from their respective leader 380.

Using the exemplary distribution methods and protocols described above in
20 reference to Figure 5A-B, the site leader 310 forms a guaranteed delivery connection 352 to subnet leaders 370, 380 using HTTP transmissions over TCP/IP connectivity. An exemplary bandwidth utilization through connections 352, 354 is approximately 100 Kbps each corresponding to, for example, a high-quality video stream. The site leader 310 may further distribute the streaming data to the
25 terminal clients 364a-c in a broadcast 356 transmitted, for example, in UDP mode. An exemplary bandwidth usage associated with this broadcast 356 is approximately 100 Kbps which is capable of being received by one or more clients residing in the same subnet as the site leader 310.

Thus, the three aforementioned primary distribution channels (connections
30 352, 354, and broadcast 356) use an exemplary total bandwidth of approximately 300 Kbps which in addition to the incoming stream received by the site leader 310 does not exceed the 400 Kbps limit. It will be appreciated that these three primary distribution channels of the streaming data are further distributed downstream in a

manner that does not significantly increase the bandwidth usage for upstream systems by replication and re-broadcast by various other subnet leaders 370, 380.

The data stream that is distributed from the site leader 310 to the subnet leader 370 of the first subnet 360 is further distributed to the four end user clients 372a-d by a UDP broadcast 374. An exemplary bandwidth usage of the broadcast 374 is approximately 100 Kbps corresponding to a single data stream. Thus, broadcasting of the data stream advantageously reaches the four receiving clients 372a-d without incurring additional significant bandwidth overhead to the subnet leader 370 for each additional client which is able to receive the broadcast information. Similarly, the data stream that is distributed to the subnet leader 380 of the second subnet 362 is further distributed to the three end user clients 382a-c by a UDP broadcast 384. An exemplary bandwidth usage of the broadcast 384 is approximately 100 Kbps, again corresponding to a single data stream. Similarly, broadcasting of the data stream advantageously reaches the three receiving clients 382a-c without incurring additional significant bandwidth overhead to the subnet leader 380 for each additional client which is able to receive the broadcast information.

As shown in the exemplary distribution configuration 350, the original streaming data from the server is distributed to a plurality of non-terminal and terminal clients (310, 370, 380, 164a-c, 372a-d, 382a-c) in the network 302. If these terminal clients were to be serviced without the efficient distribution as exemplified, each of the clients would request an independent connection to the remote server 304 to obtain the streaming data. Consequently, the bandwidth demand placed on the remote server 304 would be approximately 1300 Kbps (13x100 Kbps) which far exceeds the maximum bandwidth allocation of approximately 400 Kbps associated with the connection 342 that services the network 302. Thus, it will be appreciated that a network configured to replicate and/or broadcast a single incoming data stream advantageously permits the data stream to be received and used by a plurality of clients within the network. Furthermore, it will be appreciated that the exemplary distribution configuration 350 described above in reference to Figure 6 is one of many possible configurations utilizing the inventive concepts disclosed herein.

As previously described, Figure 6A exemplifies one embodiment of a network configuration in which broadcast data streams are used to provide streaming content to local clients and TCP/IP communications are established between leaders to distribute streaming content between different subnetworks.

5 The use of the broadcast data type in combination with the guaranteed delivery data type desirably provides a method by which the functionality of multicast-based transmissions can be substantially duplicated without the requirement of utilizing multicast-enabled hardware. In one aspect, the invention provides the ability to utilize a single stream of data transmitted from a server and replicate and
10 broadcast the contents of the stream throughout a client network in such a manner so as to reduce the amount of bandwidth that is consumed while providing transmission capability across complex network topologies and subnetworks.

Figures 6B-E further illustrate an exemplary event sequence where the streaming content distribution system is used to distribute streaming data to a
15 plurality of client computers. These figures are diagrammatically simplified to include the principal elements involved in the data streaming process. It will be understood that other hardware, routers, hubs, and other devices may be required in the actual implementation of these diagrams to distribute the streaming content between the computers shown in each figure.

20 In Figure 6B, a client network 430 is illustrated which comprises two subnets 404, 410. Each subnet 404, 410 further comprises a plurality of client devices each of which is desirably configured to receive streaming content. Each subnet 404, 410 includes a subnet leader and a plurality of clients which desirably receive streaming content broadcast by the subnet leader. Thus, the subnet 404 includes
25 a leader 406 and interconnected clients 408a-c. Similarly, the subnet 410 includes a leader 412 and interconnected clients 414a-c. In one aspect, each leader 406, 412 comprises a client device which has been elected within their respective subnet by an election process that takes into account the streaming content distribution performance capabilities of the clients in the subnet. This process is
30 described in detail in the section titled "Methods of Optimized Leader Election and Leader Feasibility Evaluation for Multiple Clients".

In an exemplary content distribution sequence, a first leader 406 residing in a first client subnet 404 is shown to receive an incoming content stream 342 from

the server 304. In one embodiment, the content stream 342 represents a multiplexed data stream encoded in a tunnel wrapper as previously described. The mode of transmission of the content stream 342 may further comprise a TCP/HTTP format which facilitates transmission of the streaming content over
5 most commonly utilized networks. The first subnet leader 406 is the entry point for the content stream 342 into the client network and therefore the first subnet leader 406 also corresponds to the site leader for the client network 430 (as shown in Figure 6A).

As illustrated in Figure 6C, after receiving the server stream 342, the leader
10 406 commences distributing the streaming content to local clients 408a-c using a subnet broadcast mode 422. As previously described, broadcasting of the streaming content in this manner advantageously permits the information contained in the data stream to reach a plurality of clients using bandwidth approximately equivalent to that required to transmit to a single client. Thus, the
15 subnet broadcast 422 simultaneously distributes the streaming content to three clients 408a-c without using three times the single-connection bandwidth. In one embodiment, the subnet broadcast 422 comprises a UDP broadcast format as previously described.

In Figure 6D, subnet-subnet communications between the first leader 406
20 and a second leader 412 occur by way of an outgoing stream 424 originating from the first leader 406. In one embodiment, the outgoing stream 424 comprises duplicated information from the incoming content stream 342. The outgoing stream 424 is further configured to be transmitted in a guaranteed delivery mode (using for example TCP/IP HTTP communications) such that the contents of the
25 data stream may be transmitted between computers residing in different subnets. Although the transmission of the outgoing stream 424 in Figure 6D is sequenced after the broadcast within the first subnet in Figure 6C, it will be understood that ordering of these events may occur substantially simultaneously or alternatively the broadcast 422 may follow the transmission of the outgoing stream 424 without
30 departing from the scope of the invention.

As shown in Figure 6E, after receiving the outgoing stream 424, the second subnet leader 412 may commence with distributing the streaming content within a second local subnet 410 by way of subnet broadcast mode 426. Broadcasting of

the streaming content in this manner reaches clients 414a-c within the local subnet 410 using only a single client transmission equivalent of bandwidth. In one embodiment, the subnet broadcast 426 takes the form of a UDP broadcast in a manner previously described.

5

Optimized Tiered Distribution of Data

As described above in reference to Figure 2, one method for improving streaming performance comprises receiving data from the remote server 240 via an encoded data stream tunneled to the client network 120. Upon receiving the encoded data stream the site leader 125 decodes the tunneled data stream and broadcasts the underlying streaming data to one or more local clients 110. This methodology advantageously alleviates bandwidth limitations by reducing the number of active connections that must be maintained by the server 240 to stream data to a client network 120.

An additional feature of the invention provides improved coordination among clients 110 of the local network 120 when processing and distributing the streaming data. In particular, specialized methodologies are implemented for efficiently distributing the streaming signal throughout the client network 120 in such a manner so as to further reduce the number of data streams required to disseminate the streaming information to each client 110 while insuring adequate available bandwidth and streaming performance. In one aspect, data streams received by a first site leader may be re-transmitted in a non-broadcast mode to other site leaders within the client network 120. This manner of transmission is useful for distributing streaming data to clients 110 which may not be able to receive broadcast transmissions from the first site leader. By relaying or retransmitting the streaming content to other site leaders, clients unavailable to the first site leader can be made to receive the streaming content when connected to another site leader that broadcasts the relayed streaming content.

Complex local networks comprising many hundreds if not thousands of client systems are typically arranged into system groupings or subnets. Each subnet comprises one or more client systems grouped together in a logical ordering. For example, using TCP/IP connectivity, a subnet may comprise a group

of computers which share a given subnet address. Each subnet may further be distinguished from one another through assignment of different subnet addresses. Subnet addressing in the aforementioned manner is often desirable as it facilitates managing complex network topologies by breaking them down into simpler groupings which can be logically arranged and administered. One possible outcome from such an arrangement is that data broadcast within a first subnet may not be recognized or "heard" within a second subnet.

When transmitting data in a broadcast manner, subnet organization may affect which clients can receive streaming content based on their position and addressing within the client network 120. Features of the invention alleviates potential problems which may arise due to complex subnet topologies by providing the ability to utilize more than one client leader to distribute streaming content. As will be described in greater detail hereinbelow, in addition to selecting site leaders that may broadcast streaming content received from a remote server, subnet leaders may further be selected which similarly broadcast streaming content received from site leaders or other subnet leaders.

In one aspect, a tiered communications architecture applies this broadcast approach and specialized methods are used to determine how the data is both relayed and broadcast throughout the local network 120. Selection of the multiple site leaders, subnet leaders, and underlying clients which connect to them is accomplished through administrative routines designed to identify the systems and/or connections which are best suited for distributing streaming content. One of these routines comprises a method for determining site leaders through performance evaluation and operational assessment to identify those clients that are better suited for receiving streaming content and subsequently distributing this content to other clients in a broadcast manner.

Another routine comprises a fan-out method that is used to determine a hierarchy or ordering of the clients to improve streaming content performance while reducing bandwidth imposed limitations and bottlenecks. The fan-out method is responsible for logically arranging the plurality of subnets within the local network so that each client has access to the streaming content provided by the server in such a manner to reduce the number of active connections which must be maintained with the server 240. Using the fan-out methodology described below, a

single connection to the content server 240 may be desirably used to distribute streaming content throughout an entire local network even if a complex network topology comprising many different subnets exists.

Figures 7A-B illustrate the incorporation of an incoming subnet into an existing streaming content architecture. In one embodiment, the streaming data connection 550 comprises a streaming server 552 and a network management server 554 collectively referred to as a server 502. The server 502 described in reference to Figures 7-10 is in fact same as the previously described servers, e.g., server 240 in Figure 2. The streaming server 552 is dedicated to transmitting the streaming content through the tunnel connection 504, while the network management server 554 is dedicated to tasks such as coordinating network connections, billing/accounting, and administrative functions. The server 502 transmits encoded streaming content to the site leader 508 through the tunnel connection 504 in a manner described above. Thereafter, the leader 508 distributes the streaming content to the existing subnet leaders in the client network.

Each subnet through which streaming content is to be distributed is connected by way of an incoming subnet leader 560. The subnet leader 560 attempts to join the existing client network where streaming content is made available by initially establishing a connection to the site leader 508. In establishing a connection the incoming subnet leader then performs a connection process 570 generally illustrated in Figure 7B. During the connection process 570 the subnet leader requests a list of available connection sources in state 574. This request may be made to the network management server 554 (as indicated by path 566), the site leader (as indicated by path 564), or other existing subnet leaders (not shown). In state 576, the incoming subnet leader 560 receives connection source information. In one embodiment, the source information comprises IP addresses or other connection characteristics for the members of the client network that are capable of distributing streaming content.

In state 580, the incoming subnet leader 560 analyzes the sources based upon the source information for various connection and source data such as bandwidth, load, routing, latency, and/or ping time. In state 582, the incoming subnet leader 560 establishes a connection with the most available source

analyzed. The most available source may be the site leader 508, or one of the other existing subnet leaders. Availability is determined in part based upon the number of active connections maintained by each source, data distribution capacity (i.e. maximum connection number), computational load, and other factors which affect the site or subnet leader's ability to distribute streaming content. Additional details of one embodiment of connecting the incoming subnet to the existing content distribution architecture is described with reference to Figure 8 below.

Figure 8A illustrates an exemplary fan-out organization 500 wherein streaming data is received by a site leader 508 from a server 502 and is thereafter distributed to clients 510 a-d within a local network 506. As previously described, the connection between the server 502 and the site leader 508 is desirably implemented using a tunneling approach wherein the tunnel connection 504 preserves the broadcastable properties of the streaming content. The site leader 508 receives encoded streaming content, using for example a TCP-based HTTP communications protocol, and may decode this information and broadcast it using, for example a UDP communications protocol. Additionally, the site leader 508 is configured to distribute the streaming content in a non-broadcast manner using a guaranteed delivery protocol such as a TCP-based communications protocol. In one aspect the TCP-based communications protocol is desirably selected to distribute non-broadcast transmissions to other subnet leaders 510 a-d. One reason for using a guaranteed delivery protocol when distributing non-broadcast transmissions is to insure that each subnet leader to which the streaming content is distributed receives a complete and uncorrupted copy of the information contained in the streaming content. Typically, the guaranteed delivery protocol provides error correcting functionality and the ability to recover from transmission errors or network latencies to preserve the integrity of the data stream as it is passed within the local network.

An exemplary local network 506 comprises a plurality of interconnected subnets (individual clients not shown). Each of the subnets comprises a subnet leader that communicates with the site leader 508 preferably using a guaranteed delivery protocol such as the aforementioned TCP-based communication protocol. Communications channels 512a-d established between the site leader 508 and

each subnet leader are utilized by the site leader 508 to distribute streaming content to each of the subnets. When establishing communication between the site leader 508 and the subnet leaders 510a-d a connection threshold or fan-out limit 507 is typically specified. The fan-out limit 507 represents a maximum
5 number of subnet clients that are allowed to simultaneously connect to the site leader 508. In one aspect, the fan-out limit desirably limits the amount of traffic or bandwidth associated with the site leader 508. Designation of the fan-out limit has the practical effect of preserving the quality of streaming content by setting a ceiling on the traffic leaving the site leader 508 to thereby reduce potential over-
10 utilization of the site leader 508. The fan-out limit 507 further prevents network performance degradation resulting from excessive numbers of subnet leaders attempting to connect to a single site leader. Additional details of the manner in which each client leader (site and subnet) is determined is described in greater detail in the section entitled "Methods of Optimized Leader Election and Leader
15 Feasibility Evaluation for Multiple Clients".

In the example shown in Figure 8A the site leader 508 is configured with a fan-out limit 507 of four connections. Thus, up to four independent connections from one or more subnet leaders 510a-d can be established before the fan-out limit is exceeded. In addition to specifying a fan-out limit for the server, bandwidth
20 limitations for each respective connection may also be specified. Thus, the fan-out limit 507 may be bandwidth dependent and dynamically change depending upon the amount of streaming content distributed by the server. It will be appreciated by one of skill in the art that the value of the fan-out limit may be flexibly defined as a static value (i.e. a fixed number of clients that are allowed to simultaneously
25 connect to the site leader) or be dynamically reassigned (i.e. a variable number based on network or site leader characteristics such as the aforementioned bandwidth limitation methodology). The value of the fan-out limit 507 is therefore responsible in part for moderating the amount of bandwidth consumed by simultaneous subnet leader connections to thereby prevent saturation of the site
30 leader.

As shown in Figure 8A, each leader (site and subnet) possesses a performance score 513. The performance score represents a value or metric which is used to measure the relative streaming content capacity of a particular

system. The performance score desirably takes into account factors such as: (1) the ability of the system to decode and process incoming streaming content (from the server or another client), (2) the capacity of the system to distribute streaming content to other clients (bandwidth or connection speed), and (3) the current load or percentage utilization of the system for performing other tasks. The performance score may be statically assigned where a single value is used to compare each systems relative performance to one another. Additionally the performance score may be dynamically assigned where each system's performance is periodically determined and updated. Dynamic assessment of system performance is useful for identifying changes in each system's relative performance which may affect their ability of distribute streaming content over time. For example, a system's performance may be significantly altered when comparing the performance of the system when it is not running any other programs or background tasks compared to when other programs are in use or network traffic is emanating from the system.

As shown in Figure 8A, a hierarchical ordering of the systems is established by comparing the performance scores for each subnet leader. Exemplary performance scores of 20, 40, 60, and 80 are associated with subnet leaders 510a, 510b, 510c and 510d respectively. Based on this information it can be determined that subnet leader 510d has the greatest performance score which generally corresponds to a system with improved capability to distribute streaming content relative to the other subnet leaders 510a-510c each of which have lower performance scores. Furthermore, the site leader 508 has a performance score of 100, higher than any of the subnet leaders' scores. In one embodiment, the site leader may in fact represent one of the subnet leaders which has the highest performance score. Alternatively, the site leader 508 may be any system within the local area network which has the greatest capacity or availability for distributing streaming content. Details of the manner in which the site leader and subnet leaders are selected is discussed in the section entitled "Methods of Optimized Leader Election and Leader Feasibility Evaluation for Multiple Clients".

In one aspect, when the number of subnet leaders requesting streaming content from the site leader does not exceed the fan-out limit, the site leader can readily distribute the content to each subnet through an independent data delivery

channel or connection. In this case each subnet leader 510 establishes a connection with the server and streaming content is distributed by the server through an appropriate communications protocol. However, when the number of subnet leaders requesting streaming content from the site leader exceeds the fan-out limit, a subnet leader rearrangement procedure may be performed to insure that the fan-out limit is not exceeded. For example, when a subnet leader 514 with a performance score of "90" makes a connection request 516 with the site leader 508, rearrangement of the existing connections 512a-d may be performed to organize the connections in a manner which insures the site leader is connected to the subnet leaders with the highest performance scores. Since the fan-out capacity of the site leader 508 is already occupied by the four subnet leaders 510a-d, the subnet leader 514 requesting connection to the site leader 508 may not be able to establish a connection with the site leader 508 unless at least one of the existing connections 512a-d is terminated or rearranged to accommodate the connection request 516. In one aspect, when an incoming subnet leader 514 makes a connection request to the site leader 508, the performance score of the incoming subnet leader 514 is compared against existing subnet leaders 510a-d already connected to the site leader 508. If the incoming subnet leader 514 possess a performance score 513 higher than any of the existing subnet leaders 510a-d then the incoming subnet leader 514 may displace one of the existing subnet leaders 510a-d. Alternatively if the incoming subnet leader 514 does not possess a performance score 513 higher than any of the existing subnet leaders 510a-d then the incoming subnet leader 514 will not displace any of the existing subnet leaders 510a-d and may instead attempt establishment under one of the existing subnet leaders 510a-d. Details of these potential rearrangements resulting from incoming site leader connection requests are described in greater detail hereinbelow in conjunction with Figure 8B.

Figure 8B illustrates an exemplary re-organization 520 which may occur when the incoming subnet leader 514 makes a connection request 516 to a site leader 508 that is connected to sufficient subnet leaders 510a-d to occupy all connections available (fan-out limit 507 is at maximum capacity). The re-organization strategy described below accommodates distributing streaming content to the incoming subnet leader 514 in such a way so as to improve

connection efficiency, reduce bandwidth bottlenecks, avoid cycles in the tiered distribution tree, and help insure that each subnet is efficiently served with streaming content.

5 In one implementation, the re-organization strategy comprises evaluating the performance scores for the existing subnet leaders 510a-d and the incoming subnet leader 514. The site leader 508 then identifies the subnet leaders with the highest performance scores and preferentially maintains or establishes communication channels as many of these systems as is permitted by the maximum value of the fan-out limit 507. As shown in the Figure, the site leader 10 508 preferentially connects to the four subnet leaders with performance scores corresponding to 90, 80, 60, and 40. Conceptually, the four selected subnet leaders 514, 510b, 510c, and 510d collectively form a second tier of subnet leaders 521. The remaining subnet leader 510a having the lower score is then arranged in a third tier 522. Here the connection between the subnet leader 510a and the site leader 508 is terminated and a new connection is established between 15 the subnet leader 510a and a new subnet leader within the second tier 521. The determination of where to connect the subnet leader 510a is again based upon identification of performance scores for each subnet leader 514, 510b-d. The displaced subnet leader 510a preferentially establishes a communications channel 20 526 with the subnet leader within the second tier 521 having the greatest performance score 513 and additionally having an available channel for communication (i.e. fan-out limit for the subnet leader is not filled to capacity).

The subnet leader 510a contained within the third tier 522 is further able to establish communication channels in a similar manner to the site leader and 25 subnet leaders contained in the second tier 521. Thus, as additional incoming subnet leaders make connection requests to gain access to the streaming content they may be arranged within a growing tree-like structure that can accommodate a large number of subnets while insuring that each leader does not exceed a designated connection capacity.

30 The re-organization strategy described above is conceived to be highly flexible and designed to permit internal subnet leader re-organization without specific connection requests by additional incoming subnet leaders. As previously described, the performance value 513 may dynamically change based on current

computational load or bandwidth availability. As a result the performance values for each subnet leader may change with respect to one another. By periodically polling the subnet leaders and comparing the current performance values at any given time, the entire subnet hierarchy can be rearranged based on the performance values. Thus, if the performance value for a third tier subnet leader increases significantly such that it exceeds the performance value for the subnet leader of the second tier to which it is connected, the connection hierarchy can be readily re-arranged by establishing new communications channels following the ordering hierarchy described above. Additionally, if a subnet leader in the second tier 521 experiences a substantial drop in its performance value such that one or more subnet leaders in the third tier 522 have performance values 513 in excess of the second tier subnet leader, the connection hierarchy can be re-arranged in a similar manner to preserve a desirably ordering of the subnets.

In one aspect, the aforementioned hierarchical ordering of subnets may also be applied to the site leader 508. In a similar manner to determining the ordering of subnet leaders in the second and third tier, the site leader 508 may be selected as the system with the highest performance score of all systems through which streaming content is to be distributed. The site leader 508 may also be dynamically re-assigned based upon changes in the performance values for both the site leader 508 and other subnet leaders within the tiered communications architecture. For example, if the performance score of the site leader 508 drops below that of an existing subnet leader, the site leader may be re-assigned as a new subnet leader and the existing subnet leader re-assigned as the new site leader. In performing such an operation, the newly assigned site leader would establish a connection with the server 502 to acquire the streaming content. Likewise, the re-assigned site leader would terminate its connection to the server 502 and establish alternative connection within the tiered communication hierarchy based upon its performance value. The dynamic re-arrangement of both site leader and subnet leader ordering in this manner maintains consistency in communications with the server and insures the streaming content is distributed efficiently.

One benefit to the aforementioned connection hierarchy is that it permits numerous clients to be able to utilize a single content stream from the server 502.

Furthermore, the connection hierarchy overcomes problems associated with broadcasting streaming content across multiple subnets by providing a means to distribute the content in a connection oriented manner. When multiple subnets exist such that a site leader 508 is not able to broadcast the streaming content along channels that can be "heard" beyond its own subnet (using for example UDP broadcasting), guaranteed delivery channels (using for example TCP/HTTP communications) are utilized to distribute the streaming content to the subnets. Each subnet leader may broadcast the streaming content to those clients within the subnet and may further relay the streaming content in a guaranteed delivery form to other subnets as necessary such that the streaming content is accessible to all clients within the local network.

When distributing streaming content it is important to accommodate temporary drop-offs or interruptions in the data stream. These interruptions may occur as a result of latencies within the network as well as switching of communications channels as the tiered communication architecture is formed and re-arranged. In order to preserve the quality of streaming content throughout the system each client system within the network is configured with a data buffer which is able to store a quantity of streaming content. The use of buffers throughout the network desirably accommodates temporary data interruptions in such a manner so as to make them transparent to the user. The use of data buffering in conjunctions with the servers, leaders, and clients of the streaming content system will be described in greater detail in connection with "Stream Splicing, De-Duplication, and Novel Buffering Layer" and Figure 15.

While the tiered communications hierarchy and network connection fan-out example depicted in Figures 8A-B is a relatively simple scenario, complex network topologies may be arranged in a similar manner. The methods described herein can be readily applied to networks containing many hundreds if not thousands of clients, significantly improving streaming content distribution within the network while reducing the load on the server 502 and intermediate network links. Furthermore, it will be appreciated that this methodology may be applied to both local and widely distributed networks and subnets.

Figures 9A-C illustrates processes 530, 531, and 532 for dynamic creation and maintenance of a desirable organization of the tiered hierarchy and network

connection fan-out. The process 530 begins in a state 540 where an incoming client connection is initiated with the intention of being added to the tiered hierarchy. The incoming client connection may arise from a new client coming online or a client whose streaming source has been interrupted for any one of several reasons described below. This client may desirably reacquire the streaming source to maintain continuity in streaming of the data. A new client connection is initiated by considering whether the client has assumed subnet leadership as described in "Methods of Optimized Leader Election and Leader Feasibility Evaluation for Multiple Clients" and exemplified in Figure 11A. In one aspect the process exemplified by Figure 11A runs in an independent execution thread and sets a state flag indicating subnet leadership status which is tested in states 542 and 544.

In state 542, a determination is made as to whether the incoming client is a subnet leader. A client which is not a subnet leader assumes the role of a non-leader client in state 541. In one aspect, leader determination is made using the leader election process described in detail in Figure 11A. More specifically, a leadership flag may be set in state 652 and this leadership flag may be cleared in state 660. Referring again to Figure 9A these flags may be used to determine the leadership state of the incoming client in state 542 to determine whether the client will assume a leadership role where it will be responsible for distributing a least a portion of the streaming content through the client network.

In the case where the client has assumed a non-leadership role in state 541 the client may be desirably configured to be the recipient of data from another subnet leader client 324 using the broadcast techniques described in "Efficient Client-side Replication, Distribution, and Broadcast". As previously described, a principle benefit to broadcasting streaming content in this manner is that substantially the same amount of bandwidth is consumed when distributing to one or more clients within the local subnet. Thus for each client connected to the server through the local subnet no substantial incremental bandwidth penalty is observed as non-leader clients do not require "dedicated" connections with the site leader but rather can "share" the same broadcast of streaming content.

In the case where the client assumes the role of a subnet leader, as exemplified by a client on a subnet with no other clients or by a client with the

highest associated performance metric on a subnet, that client leaves the state 542. The client then requests and receives 546 a list of streaming sources from the tunnel server 240. In one aspect, the list of streaming sources comprises information including the aforementioned performance metric which may be used to sort the streaming sources in order of highest performance (i.e. best metric) to lowest performance (i.e. worst metric). The performance metrics comprising information which identifies each available streaming source and their characteristics which may include performance characteristics and values, source or origination address (i.e. IP address and/or domain), type of content being requested, available bandwidth, and other information.

Because the network gateway from local area network to public or private wide area network, e.g., the public Internet, is typically the slowest and most congested link in client-server connections, it is highly desirable to reduce the number of connections and aggregate the bandwidth consumed by those connections. In one aspect, the streaming content may take the physical form of data or information transmitted using an Internet Protocol running over fiber optic cable, coaxial cable wires, telephone wires running modem or DSL protocols, satellite links, twisted pair wires, coaxial Ethernet wires, or other physical network media that support higher level network protocols. These connections are commonly referred to as last-mile connections and are typically the thinnest bandwidth links in most server delivered content. The processes of Figure 9A-C implement the desirable feature that a single server-site leader connection can be configured to provide any one site with streaming content irrespective of the number of clients associated with the site.

Referring again to Figure 9A, if the client performance metric is greater than the largest metric contained in the source list as determined in state 548, then the client assumes the role of site leader and thereafter connects to the server 550. In the case of a site leader connection, the server maintains a process 532 illustrated in Figure 9C to manage the server-site leader connections. In state 592, when a connection request is received by the server 590 from an incoming client requesting site leadership and access to the streaming content, the server confirms that the performance metric of the incoming client requesting site leadership is greater than the performance metric of the existing site leader for the

site in which the incoming client resides. If the requesting client's metric is larger than the performance metric for the existing site leader, then the requesting client is accepted as the new site leader in state 594. Site leader acceptance in state 594 is also acquired if there was no prior site leader. In the instance where there is an existing site leader, the existing site leader is requested to disconnect from the server in state 596. When disconnecting the existing site leader typically a certain amount of additional streaming data is provided to the existing site leader for use in buffering processes that are used to avoid gaps or interruptions in the streaming content. In one aspect, the amount of additional information provides to the existing site leader following the disconnection request corresponds to approximately 5 seconds of streaming content. Gap and interruption avoidance is further described in greater detail in the section entitled, "Stream Splicing, De-Duplication, and Novel Buffering Layer". Subsequently the disconnecting site leader is returned to process 530 (Illustrated in Figure 9A) at the incoming client state 540. In one aspect, upon return to this process 530 the disconnecting subnet leader is guaranteed to locate a subnet leader source other than the server because there are now 1 or more subnet leaders with higher performance metrics for that site leader to connect to.

Generally, the site leader is identified as the client with the highest performance metric amongst all clients who request access to the streaming content provided by the server. The site leader can be a dedicated system used to distribute streaming content throughout the local network or the site leader can be a multifunctional system where the site leader may present the streaming content to a user as well as distribute the streaming content to other clients within the network. In one aspect, the site leader broadcasts streaming content to clients arranged within the same subnet as the site leader, wherein the clients are capable of receiving broadcast transmissions and additionally relays the streaming content to other subnets that are not within the broadcast domain of the site leader.

In state 548, the incoming client's performance metric is compared against the list of streaming sources provided to the client in state 546. If it is determined that the incoming client has a performance metric greater than the top performing source in the list, then the incoming client assumes the role as a site leader in state 550. Proceeding to state 552 the client which has become the site leader further

assumes data service duties that are substantially identical to any other subnet leader. These duties may include operations such as receiving streaming data from an "upstream" source, and transmitting the streaming data to "downstream" clients, by the various means described herein. Typically, the subnet leader
5 communicates with other subnet leaders in the hierarchy through a guaranteed delivery protocol (TCP communications) that provides error correction capabilities and the ability to resend portions of the streaming content to insure that the information in the stream is accurately replicated. Use of the TCP protocol also provides a means to avoid many firewall and security settings in use in Internet
10 Protocol networks.

In state 544 the site leader further continues to monitor its subnet leadership status, as do all subnet leaders. If the site leader determines that it will be replaced by another subnet leader on its same subnet such that the site leader is no longer a subnet leader then the client returns to the non-leader monitor state 541. At
15 substantially the same time, whatever client that replaced the former site leader will begin executing the process 530 and will have established site leadership as a result of the new client having a higher performance metric than the prior site leader as determined by assessing the list of performance metrics for the content sources.

Returning to state 548, in the case where the client performance metric is not greater than the highest performing streaming source in the source list, the client enters a state 554 where a cycle avoidance process commences. . In one aspect, it is desirable for the tiered system to insure that no cycles be created in the developing hierarchy of clients. It will be appreciated by one of skill in the art
25 that a cycle can be represented by a general graph topology, as opposed to a tree or hierarchical topology that is desirably constructed using the system and methods described herein. In one aspect the formation of a cycle or loop in the distribution network would potentially interrupt the feed-forward characteristic of the tiered distribution topology and result in the at least two clients in the loop to
30 distribute data around in a "circular manner" without any receiving streaming content from the server.

In one aspect, a subnet leader avoids cycles by the combination of states 554, 556, 558 which take each source performance metric in turn in state 556,

from largest to smallest. When the subnet leader attempts a connection, stopping when a successful connection is achieved. State 558 implements a critical control which disallows any subnet leader from connecting to another subnet leader with a lower performance metric than itself. Thus, any client's upstream connection is to a client with a higher performance metric than itself and any path through the tiered hierarchy from the server at the "root" to the final tier of clients at the "leaves" would visit clients with successively declining performance metrics. At the "leaves" of the tree reside those clients with the lowest performance metric values. Because no client residing between a "leaf" and the server could connect to that leaf or any client in between itself and the "leaf", cycles are thereby prevented.

Another property of process 530 is that when an incoming client 540 is a subnet leader which enters the process 530, it will always either find a higher performance metric source or be the highest performance metric source itself and assume the site leader role. Thus, site leaders are always assured of having streaming sources and are always assured of avoiding cycles.

Another desirable feature resulting from the above-described process 530 is that typically no subnet leader client be required to serve more than a fixed number of clients defined by the max connections 507. Process 531 in Figure 9B describes an exemplary method for implementing this property of the client network. In one aspect, when a pre-existing subnet leader receives a request for connection in state 570, the performance metric of the requesting client and that of the pre-existing subnet leader are compared in state 572. If the requesting client has a higher score, the connection is refused in order to avoid cycles in state 574. This state 574 reflects the same cycle control that state 558 achieves. If the requesting client has a lower performance score, then the pre-existing subnet leader considers whether it is already serving its maximum number of connections in state 576. If not, the connection request can be accepted in state 580. If the subnet leader is at its maximum connections then the performance metric of the requesting client is compared to the performance metric of all the clients that the pre-existing subnet leader is serving in state 578. If the performance metric of the requesting client is less than all of the served clients, then the requesting client connection is refused in state 574.

Having been thusly refused the client connection request in state 574, the requesting client will continue to execute its cycle avoidance loop and search for an alternative source. In one aspect, additional sources are likely to exist, because, in this case, the pre-existing subnet leader serves a number of subnet leaders at the tier below itself. One or more of these lower tiered subnet leaders will be available for connections, or there will be availability at recursively lower tiers. In the case where the performance metric of the request is greater than all clients currently being served as determined in state 578 such that the requesting client has a metric higher than at least one of the clients served by the pre-existing subnet leader, the client with the lowest scoring performance metric of those served by the subnet leader is disconnected in state 582 to create an available opening for the requesting client. The disconnected subnet leader is thereafter returned to the process 530 where it will attempt to find another source in the hierarchy.

In one aspect, the tiered architecture may periodically undergo performance re-assessment wherein the performance metric for each subnet leader are re-evaluated. Any substantial changes in the performance metric for a particular subnet leader can result re-arrangement of the subnet leaders within the tiered architecture to account for the changes in the performance metrics. The manner in which the subnet leaders may be re-arranged proceeds in a manner similar to that described above. Additionally, subnet leader re-arrangement may be configured to proceed according to definable criteria. For example, the system may be configured to permit subnet leader re-arrangement based upon: (a) a subnet leader's change in performance which exceeds a designated performance threshold, (b) a designated number of subnet leaders which have undergone a performance value change, or (c) the elapse of a designated period of time. Restricting re-arrangements in the aforementioned manner may desirably prevent performance degradation due to overly frequent rearrangement of the tiered communications architecture.

It will be appreciated that the processes illustrated in Figures 9A-C may be repeated as necessary for each connection request from incoming subnet leaders. Alternatively, these processes may be invoked whenever a subnet leader exits the local network. For example, a particular subnet leader may enter a state where it

will no longer provide streaming content. In such an instance there may be a requirement to reassess the connection of the tiered communications architecture. This may include designating a new subnet leader as well as re-routing existing communications through different channels or paths within the local network. Such changes desirably occur transparently to the user and streaming content interruptions may be avoided by the aforementioned buffering methodologies.

As with other features of the multimedia distribution system and methods, the aforementioned organizational implementations and methodologies are applicable to not only multimedia streaming but to numerous other data distribution applications. Thus it is contemplated that these implementations and methodologies may be adapted for use with other data or information distribution applications without departing from the scope of the invention. Furthermore, it will be appreciated that the hierarchical subnet organization and content distribution methodology may not necessarily be restricted to operation within a single local area network. These methods may be readily adapted by one of skill in the art to interconnect multiple local area networks remotely dispersed from one another and interconnected by a communications medium such as the Internet. In one embodiment, an enterprise network may comprise different networks remotely located from one another, for example, offices in Miami, Los Angeles, and Huston. The methods described herein may be applied to these networks (and underlying sub-networks) to provide a means for coordinating content distribution in such a manner so as to reduce server load by reducing the number of active connections which must be maintained to provide streaming content to all clients within the enterprise network.

Methods of Optimized Leader Election and Leader Feasibility Evaluation for Multiple Clients

One exemplary methodology for selecting the subnet leader is now described in reference to Figures 11-12. As previously described, each subnet leader typically establishes a connection-oriented communication link with either a server 240, site leader 310, or a subnet leader 324. This communications link permits the subnet leader to reliably receive streaming content that originated from the server and was either directly delivered from the server or through one or more

tiered leaders as described in "Optimized Tiered Distribution of Data". The use of a connection-oriented communications link such as a guaranteed delivery protocol or TCP communications link desirably preserves the quality and substance of the streaming content. In one aspect the guaranteed delivery protocol performs error-correcting functions as well as provides mechanisms for recovery from network latency and interruptions. Using data buffers embedded in both the subnet leader and the site leader accommodates transitory lapses in the distribution of streaming content such that information may be presented to the user in an uninterrupted manner. Following communications establishment, the leader then broadcasts the streaming data to every client within the subnet. Broadcasting the streaming data in this manner advantageously allows the plurality of subnet level clients to simultaneously receive and utilize the streaming content while maintaining a limited number of connection-oriented communication links with the content distribution source (i.e. the site leader or the remote server).

In one embodiment, each subnet is configured such that each client within the subnet is capable of acting as a subnet leader. In order to coordinate the activities of the clients within the subnet a leader election process is active on the client. A principle function of the leader election process is to identify a single client within the subnet that is the most suited to distributing streaming content. Distribution of streaming content further comprises broadcasting streaming within the local subnet and additionally distributing streaming content to other subnet leaders or clients via guaranteed delivery protocols.

Figure 11A illustrates a leader election protocol for determining a subnet leader. A principle use of the leader election process is to provide a method for monitoring which client is the current subnet leader. Furthermore, this process provides a means for periodically establishing a new subnet leader should the performance of individual clients within the network change, new clients are added to the network, or clients are removed from the network.. According to this process each client may occupy one of three different states where the state that the client is in determines if that client is a leader. The available states comprise a suppress state 612, an announce state 614, and a listen-only state 616. When a client attempts to establish communication with a streaming content provider (i.e. a site leader or other subnet leader), it assumes that it will become the subnet leader.

This logic desirably accommodates a single member subnet and allows the entering member in the subnet to assume a leadership role for receiving and distributing streaming content. The entering client, while assuming that it will become the leader, first enters the suppress state 612 for a selected time interval
5 so as to prevent leadership conflict if a leader already exists in the subnet.

While the entering client is in the suppress state 612, it listens for any leadership declarations within the subnet. In one embodiment, the leadership declaration comprises announcements by other client's of their identity and an associated performance score. The performance score may comprise a
10 combination of metrics which define the relative performance of the client in terms of its ability to both receive and distribute streaming content. Factors which may influence the performance score include by way of example; processor speed, computational load, bandwidth capacity, latency, availability to decode and distribute streaming content, and broadcasting ability. In one embodiment, the
15 performance score takes into account these or other metrics which may be equally or differentially weighted and contribute to the determination of performance score. It will be appreciated that the performance score may be determined using a wide variety of different metrics and combinations thereof to determine relative availability of the client in performing streaming content operations it is therefore
20 conceived that different methods for computing the performance value represent but other embodiments of the system and methods described herein.

In one aspect, if no leadership announcement is received by the incoming client during the suppress time interval, the incoming client performs a leadership assertion transition 620 and enters the announce state 614. In the announce state
25 614, incoming client has assumed leadership within the subnet and periodically announces its leadership as indicated by a loop 622. The leadership announcement may comprise transmitting of signal or information to other clients within the subnet that the position of leadership is currently occupied. Furthermore, while acting as the leader, the client may receive incoming streaming
30 content from the site leader or other leaders. Furthermore, the acting leader may perform a number of streaming operations including: (a) distributing the streaming content to other subnet leaders, (b) broadcast the streaming content to other clients within the subnet, and/or (c) presenting the information to a local user.

If, however, leadership announcement is received (for example from another client currently acting as a subnet leader) during the suppress time interval, the entering client compares its own performance score with the performance score of the acting subnet leader. If the entering client's performance score is greater than the acting leader, the entering client waits the duration of the suppress interval and thereafter makes the leadership assertion transition 620 to the announce state 614. During the leadership assertion transition 620 the entering client assumes leadership within the subnet, wherein the former subnet leader with a lower performance score relinquishes leadership in a manner described in greater detail below.

If the leadership announcement from the existing leader is received during the suppress time interval, and the existing leader's performance score is greater than the entering client's performance score, the entering client makes a transition 632 to the listen-only state 616. The listen-only state 616 is similar to the suppress state 612 in that any member in either of these two states 616, 612 listen for leadership announcements from other clients, but do not themselves make any announcements. Reducing the frequency or quantity of leadership announcements required to establish and modify leadership within the subnet advantageously reduces unnecessary traffic in the subnet. In one aspect, the listen-only state 616 may be distinguished from the suppress state 612 by the fact that a client in the listen-only state 616 has relinquished leadership of the subnet, while a client in the suppress state 612 may have an undetermined leadership status.

While the entering client is in the listen-only state 616, it periodically performs a listening operation during a listen interval as indicated by a loop 626. If the entering client receives another client's announcement with a higher score during the listen interval 626, the entering client enters another listen interval upon completion of the current listen interval. If, however, the entering client does not receive a leadership announcement with a score higher than itself, it makes a transition 630 from the listen-only state 616 to the suppress state 612 upon completion of the current listen interval 626. Such a situation may arise, for example, if the current subnet leader either exits the network or has its performance score reduced by performing other tasks or operations. Once the

entering client is in the suppress state, another suppress interval elapses and during this time the entering client listens for announcements in a manner similar to that described above.

Returning to the entering client occupying the announce state 614, the manner in which it relinquishes its leadership is now described. While in the announce state 614, the entering client also listens for announcements from other clients within the subnet. For example, it may be the case that while the entering client is in the announce state 614, another client having a greater performance score transitions from the suppress state to the announce state, and thus begins to announce its performance score. The entering client then receives the announcement and acknowledges that there is a better leader in the subnet. As previously described a better leader may comprise a client that is more readily available to receive and distribute streaming content. For example, the best streaming content candidate client in a subnet may be the one that can handle the most simultaneous data streams or has the greatest distribution bandwidth. Thus, the entering client exits its current announce interval loop 622 and makes a transition 624 to the listen-only state 616. Once in the listen-only state, the entering client functions in a manner similar to that described above.

It will be appreciated that each client in the subnet is in one of the aforementioned three states, and transition between the states is exemplified by the entering client. Furthermore, the client's states may change in a dynamic manner as members enter and exit the subnet or have their performance scores change. As referred to above, the performance score for a client in the subnet comprises the combination of factors and metrics that may change over time. For example, the performance score of a particular client may change if the client is subjected to increased computation load resulting from running other programs or applications. Thus it will be appreciated that the manner in which subnet leader is elected may be advantageously configured to be flexible to accommodate the dynamic changes in performance and subnet configuration.

In one aspect, once a subnet member determines that its leadership quality is greater than that of the existing leader, it performs a process of leadership transition, of which one possible implementation is illustrated in Figure 11B. The transition process in state 672 where the new site leader contacts the streaming

source that is servicing the subnet. In state 674 that follows, the new leader commences buffering the stream from the source so as to permit a seamless continuation of the distributed stream during the leadership transition. Buffering and seamless switching of the stream is described in greater detail in the section
5 titled "Stream Splicing, De-Duplication, and Novel Buffering Layer". In state 676 that follows, the new leader coordinates with the existing leader (also referred to as the previous or old leader) to determine a transition point. The transition point may be a selected data packet number or a frame number in the stream, for example. In a query state 678, the new leader determines whether the transition point is
10 reached. For the exemplary situation where the transition point is a frame number in the stream, the new leader monitors its buffer for the selected frame number, wherein the transition point is reached when the selected numbered frame enters the buffer. If the answer in the decision state 678 is "No", the new leader continues monitoring for the realization of the transition point. If the answer is "Yes", the new
15 leader in state 680 terminates the old leader and assumes leadership of the subnet. This process terminates wherein the stream to the subnet may be subsequently distributed by the new leader.

Figure 12 illustrates an exemplary leadership determination process used by the clients as described above in reference to Figure 11A. As a client enters
20 the subnet in state 642, the incoming client enters a suppression state 644. In the suppression state 644, the incoming client remains suppressed for a selected amount of time where the incoming client does not perform any announcements to the subnet and does not assume leadership of the subnet. While suppressed the incoming client listens for leadership announcement(s) from other clients within the
25 subnet. In one aspect, listening for leadership announcements comprises awaiting and possibly receiving information from other clients within the subnet that correspond to identity information, performance values, and/or flags and data which are indicative of leadership states for the other clients. As previously
30 indicated when an incoming client enters the subnet, it assumes that it may become a subnet leader. Any performance data received from other clients while in the suppression state 644 while corresponds to leadership announcements is desirably compared against the incoming client's performance values.

Performance value comparisons are made by the incoming client in state 646 where the incoming client determines if a more qualified or available leader exists in the subnet. If in the performance value comparison state 646 the incoming client determines that the existing leader is not as qualified as the incoming client, then the incoming client waits until the selected suppress interval expires in state 650. Following expiration of the suppression interval the incoming client may assume leadership of the subnet in state 652. Once the incoming client has established itself as the subnet leader the client further proceeds to a leadership announcement state 654. During this state 654 the client performs operations associated with announcing its leadership to the subnet which may comprises broadcasting a leadership identifier to all members of the subnets as well as transmitting its performance value. Additionally, during this time the established leader continues to listen for other leadership and performance value announcements.

Periodically, the current leader enters a decision state 656 where performs operations associated with determination of whether a suitable leader exists within the subnet. If the current leader determines that its current performance values exceed those performance values for other clients then the current leader returns to state 654 where continues to function as the subnet leader.

Alternatively in decision state 656, if the current subnet leader identifies another client with greater performance values than those of the subnet leader (i.e. a more qualified leader exists within the subnet) then the member relinquishes its leadership status in state 660 and enters a listen-only state in state 662. Similarly, if the incoming client determines in decision state 646 that a more qualified leader exists, then the client enters the listen-only state 662. In the listen-only state the client may receive leadership announcements and performance values from other clients within the subnet, however during this time the client does not broadcast any leadership or performance related information. Periodically, the client in the listen-only state 662 enters a decision state 664 where the client determines if the current subnet leader is better qualified than the client itself. Existence of a more qualified leader results in the client re-entering the listen-only state 662 where it performs operations in a similar manner as described above. If the client determines that it is better suited to assume leadership than any existing client,

then the client assumes the possibility of subnet leadership in state 644 where it temporarily suppresses announcing leadership as before.

Each client desirably performs the operations related to the subnet leader election process when the client enters the network and continues to utilize this process to receive and/or transmit streaming content through the subnet. In one aspect, the process and operations of subnet leader evaluation and election comprise a functional logic which is embedded or programmed into the client system. For example, the leader election process may be implemented as a program or application run on the client computer when the client is to be desirably utilized to process streaming content.

A desirable feature of the administrative routines used in this invention is that the system improves streaming performance and quality by continuous monitoring each of the available clients which may act as the leader for receiving streaming content from the remote server or from other subnet leaders within the network. This approach not only identifies the client who is initially best suited to act as a content distributor to other clients within the subnet but also accounts for dynamic changes within the network which may affect the performance of the current leader and identify incoming clients which may be better suited for streaming content distribution. Additionally, the leader election process desirably provides a method by which a new leader can be selected if the current leader becomes unavailable or experiences significant performance decline. For example, if the current leader discontinues serving the streaming content to the clients, the leader election process may be used to identify another subnet leader which is suitable for distributing the content in such a manner so as to prevent interruptions which might otherwise be experienced by the other clients within the subnet. Thus the leader election process desirably insures that there is always a system capable of serving content to the clients of the subnet and distributing content to other subnet leaders as necessary. Furthermore, the selected leader desirably represents the most qualified system within the subnet based on available clients.

Additionally, the subnet leader election process described above with reference to Figures 11-12, advantageously permits dynamic assignment and reassignment to accommodate changes in subnet leadership which may improve

streaming content performance. Dynamic leadership assignment is a more flexible and powerful approach as compared to static assignment approaches and results in improved streaming performance by continuously attempting to ensure that the most qualified leader processes streaming content. It will be appreciated that the frequency of updating the leadership status and instructing changes in leadership may be tailored to the specific characteristics of the subnet receiving the streaming content. Thus, a network comprising many different incoming and outgoing clients may require slightly different leadership monitoring and evaluation routines as compared to a network having fewer systems which generally remain within the subnet for extended durations of time.

It will be further appreciated that the methods described herein with reference to identifying a subnet leader for one or more clients located within a subnet may be additionally applied to identification of a site leader from a plurality of subnet leaders. In this embodiment, site leader selection may utilize the aforementioned leader election methods wherein the plurality of clients are representative of subnet leaders. Site leader selection therefore may take place in a substantially analogous manner as subnet leader selection with the exception that is it not confined to operate within a single subnet.

Stream Splicing, De-Duplication, and Novel Buffering Layer

One feature of the invention relates to the ability to switch streaming sources in a manner that provides continuous content distribution with reduced playback gaps due to missing content data as compared to conventional methods. Stream jumping may be desirably used in conjunction with coordinated buffer utilization so as to reduce or eliminate content interruptions in a user transparent manner. In one aspect, stream jumping allows "switching" between two or more leaders that are distributing streaming content such that a first leader which distributes streaming content is replaced by a second leader (for example, when the second leader replaces the first leader via the election process described above in "Methods of Optimized Leader Election and Leader Feasibility Evaluation for Multiple Clients" or via the tiered delivery techniques described in "Optimized Tiered Distribution of Data") and the distribution handoff between sources is done in a manner which preserves substantially all of the data or information of the

content stream. This approach is particularly useful in real-time or live streaming applications where an interruption in the content stream may result in the loss of information. Furthermore, stream jumping enables the content distribution system to transparently switch between a first leader with a lower performance value
5 relative to a second leader with a greater performance value. By performing stream switching based on performance profiling, the content distribution system desirably maintains content stream distribution from those systems best suited for the tasks associated with receiving the content stream and subsequently redistributing it to other clients and leaders with minimal interruptions or gaps in the
10 content stream. Additionally, stream jumping may be used when a particular network route experiences high latency or becomes unavailable to preserve the quality of streaming content to the clients.

Streaming jumping also accommodates switching between content distribution leaders when a first leader is no longer used to distribute the stream or
15 if the first leader experiences an unexpected interruption in service (for example a power or device failure affecting the leader). In such an instance a second leader may resume streaming content to clients serviced by the first leader in such a manner that little or no streaming content is lost during the transition between streaming computers. In performing stream jumping operations such as those
20 indicated above, leader election and other determination processes may be used to monitor when stream jumping may be necessary. These processes are described in detail in the sections entitled "Methods of Optimized Leader Election and Leader Feasibility Evaluation for Multiple Clients" and "Optimized Tiered Distribution of Data". Figures 13, 14, and 15 illustrate exemplary stream jumping
25 processes in which the information and ordering of a content stream distributed to a client is desirably preserved. In one aspect, stream jumping utilizes content stream buffers, de-duplication processes, and splicing processes to maintain the integrity of the streaming data while providing a seamless and user-transparent switch from a first streaming content to a second streaming content source. Figure
30 16 further illustrates a generalized method of performing stream jumping based on the principles set forth in Figures 13, 14, and 15.

Figure 13A illustrates an exemplary network 700 comprising a first content provider 702 and a second content provider 704 each of which may receive

streaming content 706, 710 from upstream source(s) (not shown). The upstream source(s) may comprise one or more remote server(s) 304, site leaders 310, subnet leaders 324, or combinations thereof. In one aspect, the streaming contents 706 and 710 may be substantially similar in nature (i.e. identical content streams) or the streaming contents 706 and 710 may be substantially different in nature but share content commonalities.

In an example where similar content streams are in use and stream splicing is used to preserve the integrity of the stream, the first content provider 702 may receive a compound stream as described in "Tunneling Multiple Broadcastable Media Streams from Server to Client" with one or more types of streaming content comprising a video stream A and a video stream B. Likewise a second content provider may receive a substantially similar content stream comprising the video stream A and the video stream B. A content recipient 712 who has been receiving streaming content from the first content provider 702 may receive streaming content from the second content provider 704 in an instance where the first content provider 702 will no longer be providing the streaming content to the content recipient 712. Through stream splicing, the streaming content comprising video stream A and video stream B will be desirably preserved though the client providing the streaming content to the content recipient has "switched" from the first content provider to the second content provider.

In another example, streaming content partially different nature may be received by the first content provider 702 and the second content provider 704. The first content provider may receive one or more types of streaming content comprising an audio stream A, a video stream A, and a video stream B, likewise the second content provider may receive one or more types of streaming content comprising three video streams A, B, and C. When performing streaming operations including those related to stream splicing, either a portion of the streaming content or the streaming content in its entirety may be transmitted between clients. In the case of stream splicing, the content recipient 712 may receive video stream A and video stream B from the first content provider 702. As will be described in greater detail hereinbelow, the second content provider 704 may be used in stream splicing operations to provide the content recipient 712 with

substantially the same contents of the stream that was being received from the first content provider 702.

The methods for stream splicing described herein are not necessarily protocol dependent and as such connectionless protocols such as UDP communications protocols may be applied to the splicing processes. It is generally desirable however, to utilize a guaranteed delivery protocol such as a TCP communications protocol to permit error correction and resend capabilities which facilitate maintaining stream integrity. Furthermore, the application of stream splicing methodologies may be desirably used in connection with streaming data distributed by fan-out communications methodologies and by broadcasting methods as described in greater detail in the sections entitled "Optimized Tiered Distribution of Data", "Methods of Optimized Leader Election and Leader Feasibility Evaluation for Multiple Clients", and "Tunneling Multiple Broadcastable Media Streams from Server to Client" respectively.

Figure 13B illustrates an exemplary application of stream splicing 720 as it applies to the first content provider 702, the second content provider 704, and the content recipient 712. As shown in the illustration the first content provider 702 terminates its service of providing the streaming data to the content recipient 712, for example, due to termination of its connection (706 in Figure 13A) to the upstream source, a power interruption, a device failure, a change in performance, or other event which may lead to discontinuation of transmission of the streaming content from the first content provider 702. The content recipient 712 obtains a new feed of streaming data 722 from the second content provider 704 such that the transition is substantially seamless and desirably maintains the information in the content stream in a complete and uncorrupted form. Figure 14 illustrates one embodiment of the time dependency of a stream jumping sequence 730 in terms streaming content transmission state for the first content provider 702 transmitting the first data stream 714 and the second content provider 704 transmitting the second data stream 722 (shown in Figures 13A-B above). In the stream jumping sequence, the convention of a "high" state refers to an active state and a "low" state refers to an inactive state. The sequence 732 represents a time interval where the first content provider 702 is actively transmitting the streaming to a content recipient 712 content up to a time T1, where the first content provider

transitions from transmitting the streaming content (service "on") to an inactive transmission state (service "off"). When such a transition or termination of streaming content is detected or anticipated within the local network, the content recipient 712 may execute a "jump" to another source of streaming content. In one embodiment, the stream jumping involves communication between the first content provider 702 and the content recipient 712 wherein the first content provider 702 notifies the content recipient 712 of the impending discontinuation of transmission of streaming content by the first content provider. Upon notification, the content recipient 712 identifies the current type of streaming content being transmitted (i.e. what channels are being transmitted) and/or the current position or data (i.e. the time index or the data near the point of termination). Using this information the content recipient 712 acquires a new source as described in "Optimized Tiered Distribution of Data". Content recipient 712 requests the appropriate streaming content to be transmitted as well as the exact information which must be sent in order to preserve the integrity of the content stream by initiating a connection to the second content provider 704.

At time T2, the content recipient may become "latched" to the second content provider as the streaming content provider. In one aspect, latching takes the form of the content recipient 712 establishing a connection with the second content provider 704 (for example establishing a guaranteed delivery connection using TCP communications). Alternatively, in the case of broadcast type transmissions, the content recipient 712 may gain knowledge of the channel or port that the second content provider 704 will be broadcasting on and listen in to this channel or port for streaming information (this communications form may use protocols such as UDP which is the Internet Protocol used for broadcast transmissions). As indicated by the sequence 732 and 734, the second content provider 704 is in an "off" state with respect to the content recipient 712 prior to time T2 and in an "on" state after time T2. In one aspect, the "off" state represents the second content provider 704 not maintaining an active connection with the content recipient 712 or alternatively the content recipient 712 may not be tuned in to the transmissions of the second content provider 704 during the time interval up to T2.

In general, the events associated with time T2 proceed at some point after time T1. This is especially true if the first content provider 702 terminates service with the content recipient due to an unexpected interruption in content transmission (for example from a power failure, network routing breakdown, device failure, etc.)

5 Additionally, the first content provider 702 may anticipate a service interruption and when transmitting streaming content to the content recipient 712 and thus notify the content recipient. The stream jumping sequence 730 may, however, be arranged such that streaming content transmission transition between the first content provider 702 and the second content provider 704 is overlapping wherein

10 time T2 occurs earlier than time T1. Such an occurrence may be represented when the content recipient 712 is capable of receiving streaming content from both the first and second content providers simultaneously. Additionally, at least a portion of the streaming content transmitted by the first and second content provider during this time may be duplicated or redundant, in which case the

15 content recipient is able to distinguish between the two transmission sources and disregard the duplicated streaming content.

The stream jumping sequence 730 further comprises a sequence 736 representative of the state of the first content stream originating from the first content provider and utilized by the content recipient 712. As indicated by

20 sequence 736, the first data stream may be terminated at time T3 that occurs some time after time T1 when the first content provider's servicing of the content recipient streaming content requirements has terminated. In one embodiment, a client that will discontinue the service of streaming content is configured to continue to send out the streaming data for a selected amount of time after it has

25 ceased to be the servicing client. In another embodiment, the discontinuation of service of streaming content may proceed without content interruption as a result of buffered data which may be sent to the content recipient prior or after the first content provider has ceased servicing the content recipient. Here the buffered data desirably provides a means to continuously provide streaming content without

30 interruption by creating a bridge between the content transmitted by the first content provider and the content transmitted by the second content provider. During the time of transition between the content servers, the buffer allows previously stored data to be streamed within the content recipient in a manner

transparent to the user. The second content provider 704 desirably connects to the content recipient 712 before the buffer has been exhausted and resumes streaming content transmission in such a manner that the content in the buffer is synchronized with the content of the transmitted data stream. Furthermore, the data may be streamed to the content recipient in such a manner so as to again fill the buffer such that other delays in stream jumping can be tolerated without perceptible service interruption.

Sequence 738 represents the acquisition of the content stream 722 transmitted by the second content provider to the content recipient 712 that is initiated at time T4. In one aspect, time T4 occurs some time after time T2 where the second content provider 704 has started serving the content recipient 712 with the streaming content. The buffer present in the content recipient 712 is capable of accommodating long delays in reacquisition of the data stream corresponding to the transmitted content. The delay between time T4 and T3 is desirably reduced wherever possible or practical to reduce the size of the buffer required to transition from the first data stream to the second data stream. In one aspect, transitions times in the millisecond range are easily obtainable which are conveniently accommodated by the buffer which may store 5-25 seconds of streaming content or more.

The stream jumping sequence 730 illustrated in Figure 14 is in one possible configuration wherein the second content provider assumes the role of active content distributor at time T2 which may be after the first content provider has terminated its transmission of streaming content at time T1. Furthermore, the stream jumping sequence 730 may be configured such that a lag exists between the time when a content distribution client changes state and the time when its corresponding content stream also changes state. Specifically, time difference T3-T1 is indicative of the lag between the termination notification of the first data stream and termination of actual data transmission from the first content provider 702.

In Figure 14, the time of transition of second data stream at T4 is arbitrarily positioned relative to time T3 representing the end of the first data stream. In certain stream jumping situations, time T4 may occur later than time T3. In such situations a time gap exists between acquisition of the two data streams. In other

stream jumping situations, T4 may be less than T3, resulting in an overlap in acquisition between the two data streams. In both of these situations the corresponding gap and overlap in streaming content acquisition are advantageously managed by operations including splicing, de-duplication, and
5 buffering of the data streams in a manner described below.

Figures 15A-F illustrate various "snapshots" of a buffer 742 in the content recipient as it performs the stream jumping from the first content provider to the second content provider (illustrated above in reference to Figures 13-14). At time T1, the contents 740 of the buffer 742 comprise a plurality of frames of streaming
10 data, wherein data stream 744 is the input into the buffer 742 having being transmitted or broadcast from a stream server, site leader, or subnet leader. In one aspect, the data stream 744 which is input into the buffer comprises streaming audio or video information that has been processed and formatted by the stream server or subnet leader for user viewing and/or listening depending on the type and
15 content of the stream. As previously described, the streaming content may comprise numerous types and formats of data corresponding to industry recognized formats such as AVI, MPEG, QT, and other that are recognizable and decodable by commercially available software packages. These commercially available software packages may be configured to work in corporation with a
20 dedicated software package or streaming client application that is used to implement the various features of the invention. It will be appreciated that the various features of the invention provide improved data transmission functionality and may be used not only with streaming content applications but in other applications where bandwidth reduction, performance assessment, and load-
25 balancing are important to maintain quality and consistency in data transmissions across a network.

An output stream 746 is further illustrated as emerging from the buffer 742. In one aspect, the output stream 746 comprises the streaming content which has been appropriately decoded, processed, verified and is ready for interpretation by
30 the hardware or software component used to present the streaming content to the user. The streaming client application which performs the stream splicing operations may further be configured to interface directly with the hardware or software component used to present the streaming content to the user.

The frames within the buffer comprise packetized information which in one aspect may take the form of streaming content coded in the form of (Internet Protocol) IP packets. Each frame should be ordered in a particular manner such that when the information that is contained within the frames is played back through a suitable application the streaming content can be resolved (i.e. music played or video watched). The size of the buffer which contains the frames is not necessarily restricted to a particular size and may be configured to store packetized information corresponding to a few seconds of streaming content up to a few minutes of information or more. In one preferred embodiment, the size of the buffer is configured to accommodate between approximately 5 seconds to 25 seconds of information. The buffer 742 may be implemented by way of example, using a portion of the client's memory or a portion of a storage device accessible by the client.

An exemplary frame 748, termed the I-th frame, is shown as a visual reference for the purpose of illustrating how streaming content passes through the buffer. It will be understood that progression of the streaming content through the buffer in a serial manner is for illustrative purpose only and is to be generally analogous to the time progression of stream jumping described above in reference to Figure 14. Furthermore, the plurality of frames of the streaming data in the buffer may be arranged manipulated in any number of ways without departing from the spirit of the invention. At time T1, the first content provider may be in the process of terminating its connection to the content recipient such that streaming content will no longer be received from the first content provider. At this point, the I-th frame has entered the buffer of the client and may be followed by one or more additional frames of information.

At time T2, the first content provider is no longer transmitting streaming content to the content recipient and at this point the second content provider begins to transmit streaming content at substantially the same point where the first content provider terminated its distribution of streaming content. During the time it takes for the second content provider to commence transmitting the streaming content, the content recipient may be actively utilizing the streaming information. As a result the contents of the buffer may be continuously consumed such that the I-th frame may be nearing use by the streaming content application. A principle

feature of the stream splicing system and methods described herein is to ensure that the buffer does not become depleted such that the data stream runs out sometime after the I-th frame. Should such an event occur the buffer would become depleted and an interruption in the streaming content would be observable by the user as a drop-off or interruption in the currently playing content stream. In another aspect, the difference in time between the loss of acquisition of the first data stream and the acquisition of the second data stream creates a lag in transmitted frames received by the buffer. For the purposes of illustration, the contents 750 of the buffer 742 are shown at time T2 and comprises time shifted segment of the streaming content where the I-th frame 748 is shifted to the right indicating its position in the buffer has changed as it nears being used by the streaming content presentation application.

At time T3, the first data stream 744 terminates at frame N 754 with this frame being the last frame that is received by the content recipient from the first content provider following termination at time T1. At this point the content stream is broken with the contents 752 of the buffer comprising the last received portion of the data stream 744 corresponding to the frame N 754. Though the content stream is broken at this point, a user will not immediately recognize the break as an interruption because of the amount of streaming content buffered in front of the frame N 754. Therefore if the break in the stream can be identified and the broken stream rejoined to another stream from this point on, then the user will not perceive an interruption in transmission because the stream will have been "repaired" prior to the usage of frame 754 by the streaming content presentation application. In this way the buffer acts to repair broken data streams in a user transparent manner that desirably to not affect the output quality of the content stream.

As referred to above in reference to Figure 14, the point in time T4 where the second data stream commences transmission may be either greater or less than time T3. As such, both situations are illustrated and described in reference to Figure 15 below. In one instance, the terminal portion of the first data stream comprises, in decreasing time order, frame N and frames N-1, N-2, etc. that duplicate frames already received from the first data stream. Sometime thereafter the buffer 742 receives the portion of the second data stream 758 corresponding to the duplicated content of frame N and consecutive streaming frames N+1, N+2,

N+3, etc. Furthermore, the second data stream may further comprise one or more additional frames N-1, N-2, etc. the temporally reside before the frame N. Thus, in the instance where stream switching occurred such that time T4 is greater than time T3, an exemplary buffer content 756 comprises frames N-2 through N+1 for the second data stream 758 and frame N-2 through frame N 754 of the first data stream.

By analyzing the frames in the buffer arising from the first data stream and the second data stream a stream splicing application identifies which of the frames in the buffer content 756 are duplicated. Identification of the duplicate frames is useful for determining the position of the overlap in the two data streams. Frames in the position of overlap can be used to determine the position where the stream splice should take place. Furthermore, the stream splicing application may identify the last frame to be transmitted in the first data stream as the final opportunity to splice the information from the two data streams together without losing information or streaming content. As shown in the buffer content 756, frame N 754 from the first data stream is duplicated in the second data stream. Furthermore, frames N-1 and N-2 of the second data stream, may also be recognized as duplicate copies of the same numbered frames in the first data stream. Based on this information, a splice 762 may be made between frame N of the first data stream and frame N+1 of the second data stream. In joining of the streams in this manner the content of the data stream is preserved in its original unduplicated form while the stream is still proceeding through the buffer where it will later be used by the streaming content application for presentation to the user.

In one aspect, the stream splicing process additionally performs a de-duplication operation where identical frames in the first data stream and the second data stream are identified and one set of frames is removed to create a single contiguous data stream. De-duplication in this manner is important to insure the duplicate frames are not passed to the streaming content application. Should duplicate frames reach the streaming content application, a number of undesirable consequences may result. For example, the streaming content application may recognize the duplicated frames as a streaming error resulting in interruption of the streaming content or crashing of the application. Furthermore, the streaming content application may process both frames consecutively with unexpected

results which may negatively impact the quality of the streaming content. Additionally, if duplicate frames are not identified and removed, the content stream may become corrupted and passed to other clients within the local network further causing problems.

5 As shown by way of example in Figure 15D the de-duplication operation of the stream splicing process has the effect of removing duplicate frames N, N-1, and N-2 of the second data stream while leaving the contents from these frames which originated from the first data stream intact. At this point a stream splice operation 762 can be performed where frame N of the first data stream is joined
10 with frame N+1 760 of the data second data stream 758 to yield a contiguous data stream containing only one copy of each valid frame of the streaming content. Alternatively, instead of removing duplicate frames N, N-1, and N-2 from the second data stream, frame N could be removed from the first data stream (the last available frame from the first data stream) and joined with the second data stream
15 following the removal of frames N-1 and N-2. It will be appreciated that these exemplary identification of a stream joining regions and operations are but a few examples of how the stream splicing process function to preserve the integrity of the streaming content and as such other stream and frame compositions may be used without departing from the scope of the invention.

20 For the case of T4 less than T3, exemplary buffer content 766 in Figure 15E comprises frames N-1 to N-5 of the first data stream 744, and frames N-2 to N+2 of the second data stream 758. Again, the frames N-2, N-1, and N are common to both of the first and the second data streams and thus one of each the frames may be desirably removed while in the buffer to generate the contiguous streaming
25 content.

 In the exemplary situation depicted in Figure 15E, the second data stream 758 commences significantly earlier than the end of the first data stream such that there is a large amount of overlap between the frames of the first and the second data streams. In such an instance, portions the second stream may be
30 discarded until the termination of the first stream has been detected. At this point, stream joining operations may be used to form the single contiguous data stream.

 Finally, Figure 15F illustrates the single contiguous data stream which leaves the buffer after time T4, wherein buffer content 778 comprises frames N+2

to N+6 which originates from the second data stream 758. The buffer output stream 746 now advantageously comprises frames N+1, N, N-1, etc., wherein a seamless continuation in the output data stream is made at frame boundaries 780 ($T4 > T3$) or 782 ($T4 < T3$).

5 A further feature of the stream splicing process is that these operations may be used in broadcast operations of simultaneous two or more streams for the purposes of error correction and preservation of stream integrity. In one aspect, broadcast protocols such as UDP communications do not provide error correcting functionality. As a result, during streaming operations one of more frames may not
10 be received or be corrupted in transit from the server or site leader to the client. By receiving simultaneous broadcasts from two or more stream sources, the client can make use of the buffer, stream splicing operations, and de-duplicating operations to improve the quality of the streaming content. In one aspect, the client may receive two independent communications streams of the same streaming content.
15 The client may then apply stream comparison routines to ensure that all frames have been received. If a frame is missing or corrupted its contents can be acquired from the other independent communications stream to re-form a more accurate representation of the original stream. Other duplicated frames are removed from the stream buffer by the aforementioned de-duplication processes.

20 Various parameters associated with stream jumping, such as timing of the buffering and splicing operations may be configured in a number of ways. In one embodiment, the first content provider is aware that a stream jumping event will occur and is configured such that it continues to transmit the first data stream for a selected time interval so that the first data stream overlaps with the second data
25 stream transmitted by the second content provider. In applications where the streaming content corresponds to streaming video or audio, the stream transmission interval prior to the first content provider terminating its streaming content transmission may correspond to from anywhere less than a second to several seconds or more of the streaming content. It will be appreciated that these
30 aforementioned transmission time intervals, as well as other parameters associated with stream jumping, may be advantageously configured in an optimal manner depending on other characteristics of the streaming content system and

applications (for example, the capacity or size of the buffers present within the clients).

The stream jumping exemplified above in reference to Figures 13-15 is now summarized in reference to Figures 16A-B. Figure 16A illustrates a stream jumping process 790 that is implemented to switch between servers or subnet leaders that provide content distribution. In state 791, a client receives streaming data from a first content distribution source (i.e. a server or a subnet leader). At some point thereafter the receiving client receives a notice in state 792 indicating that the first content distribution source will be terminating its service and streaming content will become unavailable through the first content distribution source. Upon receiving the termination notice, the client selects an alternative second streaming content distribution source in state 794. The second distribution source may be identified by the client directly as described in "Optimized Tiered Distribution of Data" or information may be passed from the first content distribution source which provides the client with appropriate connection information to be used in connecting to the second content distribution source. Alternatively, content source distribution may be determined by a leader election process which automatically determines who the second content distribution source will be based on performance characteristics of other clients within the subnet. The leader election process is described in detail in the section entitled "Methods of Optimized Leader Election and Leader Feasibility Evaluation for Multiple Clients".

After identifying the second content distribution source in state 794, the process proceeds to a state 796 where the client performs the stream splicing operation wherein the streaming content provided by the first content distribution source is joined with the streaming content provided by the second content distribution source in such a manner so that a continuous data stream is produced without loss or duplication of the content. As previously indicated stream splicing in this manner provides a method by which the client may utilize two or more different sources for streaming content while maintaining consistency in the data stream such that the operations take place in a user transparent manner.

Figure 16B further illustrates the streaming data source switching process 800 that occurs in state 796 of the stream jumping process 790 described above in

reference to Figure 16A. In one embodiment, the switching process 800 operates by periodically monitoring the status or availability of the streaming data sources. In state 802, the switching process 800 performs routine data streaming operating such as buffering the incoming data stream prior to its use by a streaming content application residing on the client. In a decision state 804, the process determines if a stream jump is necessary to switch between streaming content sources (i.e. the first content source and the second content source). In one aspect, determination of an impending stream jump occurs when a client receives notification from the first or second content server that the first content server will be terminating its transmission of the content stream. Alternatively, the client may determine a stream jump is necessary based on an interruption in the content stream arising from the first content server. If in state 804, the client determines that no stream jump is pending or necessary then the process resumes its routine operations of stream reception and stream buffering in state 802. If in state 804, the client determines that a stream jump is necessary, then the process proceeds to a state 806 where the client simultaneously buffers portions of the first data stream arising from the first content source and the second data stream arising from the second content source. In state 808 that follows, the process 800 determines the temporal orientation of the two streams to determine a point where the first and second data stream may be synchronized based on one or more duplicate data packets or frames. Upon synchronizing the streams the process 800 identifies all duplicate information within the two streams and removes the duplicated information from one or the other data streams contained within the buffer. In state 810, the process 800 joins or splices the information contained in the two data streams so as to obtain the original packet or frame sequence in the output data stream. Thereafter the process then resumes its routine buffering operations in state 802 and awaits the next stream jumping event.

It will be appreciated that the various aspects of stream jumping described above advantageously provide seamless data streaming when using two or more content distribution sources. Seamless transitions in content streaming are highly desirable in applications such as streaming video and audio content, wherein any missing or duplicated packets or frames may result in noticeable interruptions or

degradation in the quality of the that resultant content that may detract from user's viewing or listening enjoyment.

WHAT IS CLAIMED IS:

1. A method for encoding multimedia information for distribution across a communications network to a plurality of client devices, the method comprising:
 - receiving the multimedia information in the form of at least one data stream from a stream source;
 - performing an operation to transform the data stream into a broadcastable data stream;
 - combining the at least one broadcastable data stream into a multiplexed data stream; and
 - encoding the multiplexed data stream in a tunnel wrapper to generate an encoded multiplexed data stream that can be transparently transmitted across a network.
2. The method of Claim 1, wherein the at least one data stream comprises at least one formatted data stream.
3. The method of Claim 2, wherein the at least one formatted data stream is selected from the group consisting of: a TCP formatted data stream, a broadcast formatted data stream, a UDP broadcast data stream, a point-to-point data stream, a connection oriented data stream, a guaranteed delivery data stream, a unicast formatted data stream, and a UDP unicast data stream.
4. A method for distributing streaming content across a communications network to a plurality of client devices, the method comprising:
 - receiving the streaming content in the form of a plurality of raw data streams from at least one stream source;
 - converting the plurality of raw data streams into a multiplexed data stream;
 - encoding the multiplexed data stream in a tunnel wrapper to generate an encoded multiplexed data stream that can be transparently transmitted across a network;
 - transmitting the multiplexed data stream to at least one client;
 - receiving the multiplexed data stream at the client and subsequently decoding the encoded multiplexed data stream to generate the multiplexed data stream; and

broadcasting the streaming content corresponding to one or more of the raw data streams contained in the multiplexed data stream using independent broadcast channels.

5 5. The method of Claim 4 wherein the tunnel wrapper is configured to permit transparent transmission of the multiplexed data stream in a manner so as to preserve an underlying format and content of the multiplexed data without direct interpretation by the communications network.

10 6. A method for distributing streaming content across a non-multicast enabled communications network to a plurality of client devices, the method comprising:

 receiving the streaming content in the form of at least one raw data stream from a stream source;

 converting the at least one raw data stream into a presentation compatible data stream;

15 performing a local multicasting operation to transform the formatted data stream into a broadcastable data stream;

 combining the at least one broadcastable data stream into a multiplexed data stream;

20 encoding the multiplexed data stream in a tunnel wrapper to generate a encoded multiplexed data stream that is transmitted across the non-multicast network; and

25 receiving and decoding the transmitted encoded multiplexed data stream by at least one client device which acts as a leader to further distribute the streaming content contained in the multiplexed data stream to the client devices in a broadcast mode.

30 7. The method of Claim 6, wherein the formatted data stream is selected from the group consisting of: a TCP formatted data stream, a broadcast formatted data stream, a UDP broadcast data stream, a point-to-point data stream, a connection oriented data stream, a guaranteed delivery data stream, a unicast formatted data stream, and a UDP unicast data stream.

8. The method for distributing streaming content of Claim 6, wherein:

 the leader transmits the streaming content contained in the multiplexed data stream using independent broadcast channels

corresponding to the streaming content originating from each stream source.

9. The method for distributing streaming content of Claim 6, wherein:

5 the leader further distributes the streaming content contained in the multiplexed data stream to the client devices using a guaranteed delivery protocol.

10. A method for streaming multimedia data generated by a plurality of content sources over a communications network, the method comprising:

10 transforming the multimedia data generated by each content source into a plurality of formatted datastreams;

converting the plurality of formatted data streams into a plurality of broadcastable data streams;

packaging the plurality of broadcastable datastreams into a multiplexed data stream; and

15 tunneling the multiplexed data stream through a communications network to at least one client device wherein the client device subsequently recognizes the plurality of broadcastable datastreams and transmits the multimedia data contained therein to at least one client device in a broadcast mode.

20 11. The method of Claim 10, wherein the plurality of formatted data streams are selected from the group consisting of: TCP formatted data streams, broadcast formatted data stream , UDP broadcast data streams, point-to-point data streams, connection oriented data streams, guaranteed delivery data streams, unicast formatted data streams, and UDP unicast data streams.

25 12. The method for streaming multimedia data of Claim 10, wherein:

the multimedia data transmitted in the broadcast mode utilizes a UDP broadcast data type.

13. The method for streaming multimedia data of Claim 10, wherein:

30 upon recognizing the plurality of broadcastable datastreams the client transmits the multimedia data contained therein to at least one client device in a guaranteed delivery mode.

14. The method for streaming multimedia data of Claim 13, wherein:

the multimedia data transmitted in the guaranteed delivery mode utilizes a TCP/IP data type.

15. The method for streaming multimedia data of Claim 13, wherein:

5 the multimedia data transmitted in the guaranteed delivery mode utilizes a HTTP data type.

16. A method for streaming data comprising a plurality of channels comprising multimedia content from a server to a plurality of clients, the method comprising:

10 transforming the streaming data for each of the plurality of channels into broadcastable streaming data types at the server and further combining the broadcastable streaming data types into an encoded multiplexed datastream;

15 transmitting the encoded multiplexed datastream across a communications network to a least one of the plurality of clients which acts as a site leader; and

decoding the multiplexed datastream by the site leader and thereafter re-transmitting at least one of the plurality of channels of streaming data in a broadcast mode to clients residing in the same subnet as the site leader.

20 17. The method of Claim 16, wherein the site leader resides on the same subnet as the clients and wherein the broadcast mode comprises a UDP data type.

18. The method of Claim 16, wherein the site leader further re-transmits at least one of the plurality of channels of streaming data using a guaranteed delivery mode which provides streaming content to clients residing outside of the subnet in which the site leader resides.

25 19. The method of Claim 16, wherein the plurality of channels of streaming data are obtained from at least one content source.

20. The method of Claim 19, wherein the content source is selected from the group consisting of: a satellite transmission signal, a land based broadcast transmission signal, a video transmission signal, and an audio transmission signal.

30 21. The method of Claim 16, wherein the broadcastable streaming data type is a multimedia stream.

22. The method of Claim 16, wherein the multimedia stream has a format selected from the group consisting of: a Windows Media format, a Moving Picture

Experts Group (MPEG) format, an Audio Video Interleave (AVI) format, a Real Media (RM) format, a QuickTime (QT) format, and a MPEG Audio Level 3 (MP3) format.

23. The method of Claim 16, wherein the broadcastable streaming data type
5 is encoded using a commercially available stream encoder.

24. The method of Claim 23, wherein the encoder is selected from the group consisting of: a Microsoft Windows Media Encoder, a Xing MPEG Encoder, and a QuickTime Encoder.

25. The method of Claim 16, wherein the multiplexed datastream is encoded
10 using a tunneling protocol to generate the encoded multiplexed datastream that is compatible with the communications network.

26. The method of Claim 25, wherein the communications network comprises the Internet.

27. The method of Claim 16, wherein the encoded multiplexed datastream is
15 transmitted using a TCP/IP communications protocol.

28. The method of Claim 16, wherein the encoded multiplexed datastream is transmitted using a Hypertext Transfer Protocol (HTTP).

29. A method distributing streaming content from a server to a plurality of clients in such a manner to preserve available bandwidth, the method comprising:
20 designating at least one of the plurality of clients to act as a site leader wherein the site leader communicates with a remotely located server through a communications medium;
transmitting the streaming content from the server to the site leader;
configuring the site leader to receive the streaming content and re-
25 transmit at least a portion of the streaming content over a first subnetwork in a broadcast mode, the site leader client further configured to transmit the streaming content to a subnet leader client associated with a second subnetwork in a guaranteed delivery mode wherein the subnet leader client re-transmits at least a portion of the streaming content over the second
30 subnetwork in a broadcast mode;
configuring a first plurality of clients residing within the first subnetwork to acquire the streaming content re-transmitted over the first

subnetwork by the site leader wherein the broadcast mode provides shared accessibly to the streaming content; and

configuring a second plurality of clients residing within the second subnetwork to acquire the streaming content re-transmitted over the second subnetwork wherein the broadcast mode provides shared accessibly to the streaming content.

5 30. A method of distributing streaming content within a client network that comprises a site leader client and at least one additional clients and wherein the streaming content originates from a remotely located stream server, the method comprising:

transmitting the streaming content from the remotely located stream server to the site leader; and

receiving the streaming content by the site leader and subsequently replicating the streaming content for distribution to at least one client located in a local subnetwork using a broadcast transmission mode and further replicating the streaming content for distribution to at least one client located in a remote subnetwork using a guaranteed delivery mode.

15 31. The method of distributing a streaming content of Claim 30, wherein the at least one client located in the remote subnetwork acts as a subnet leader to receive the streaming content distributed using the guaranteed delivery mode and thereafter retransmits the streaming content in a broadcast mode to at least one client located in the remote subnetwork.

20 32. The method of distributing a streaming content of Claim 30, wherein server bandwidth is preserved by replicating and retransmitting the streaming content received from the server such that each of the plurality of clients are provided with the streaming content using a single connection between the server and the site leader.

25 33. The method of Claim 30, wherein the site leader is selected from the plurality of clients on the basis of a performance metric that is used to evaluate each available client and determine a suitable site leader that may be configured to receive the streaming content from the remotely located server.

34. The method of Claim 30, wherein the streaming content transmitted from the remotely located server comprises a tunneled data stream that provides network transparency through the communications medium.

5 35. The method of Claim 34, wherein the tunneled data stream transmitted from the remotely located server utilizes a TCP/IP transmission protocol.

36. The method of Claim 35, wherein the TCP/IP transmission protocol further comprises a Hypertext Transfer Protocol (HTTP).

10 37. The method of Claim 30, wherein the streaming content comprises a multiplexed data stream further comprising at least two discrete channels of streaming content.

38. The method of Claim 30, wherein the site leader resolves the at least two discrete channels of streaming content from the multiplexed data stream and selectively distributes the streaming content associated with the discrete channels.

15 39. The method of Claim 30, wherein the guaranteed delivery mode comprises a TCP/IP transmission protocol.

40. The method of Claim 30, wherein the guaranteed delivery mode is used to provide connection-oriented communications which support error correction.

41. The method of Claim 30, wherein the guaranteed delivery mode comprises a Hypertext Transfer Protocol (HTTP).

20 42. The method of Claim 30, wherein the site leader is further configured to present the streaming content to a local user.

43. The method of Claim 30, wherein the distributing member receives and relays the data stream to distributing members by connection-oriented communication.

25 44. The method of Claim 30, wherein the streaming content comprises a multimedia data stream.

45. The method of Claim 30, wherein the streaming content comprises an audio component.

30 46. The method of Claim 30, wherein the streaming content comprises a video component.

47. A method for distributing an incoming data stream within a client network wherein the incoming data stream is received from a server outside of the client

network and wherein the client network comprises a plurality of clients distributed across at least one subnetwork, the method comprising:

identifying one or more clients that have requested access to the incoming data stream;

5 analyzing the clients on the basis of a performance metric wherein the performance metric is indicative of each client's ability to receive and distribute the incoming data stream;

forming a distribution arrangement based on the performance metric wherein the client with the highest performance metric is designated as a site leader and wherein at least one client with a lesser performance metric located outside of the subnetwork where the site leader is located is designated as a subnet leader which establishes a guaranteed delivery connection to the site leader; and

15 configuring the site leader to receive the incoming data stream wherein the site leader replicates the contents of the incoming data stream and broadcasts the data stream contents to clients located in the same subnetwork as the site leader and furthermore the site leader replicates the contents of the incoming data stream and transmits the replicated data stream to the subnet leader through the guaranteed delivery connection.

20 48. The method for distributing streaming content of Claim 47, wherein the analysis of performance metrics and subsequent arrangement of clients results in improved distribution efficiency of the incoming data stream.

49. The method for distributing streaming content of Claim 47, wherein the method efficiently distributes the contents of the incoming data stream to each client requesting access to the data stream while maintaining a single connection to server.

50. The method of Claim 47, wherein the performance metric which undergoes analysis is selected from the group consisting of: the computational speed of each client, the computational load of each client, the bandwidth capacity of each client, and the network load of each client.

51. A system for replicating and broadcasting a data stream in a client network wherein streaming content is transmitted from a server and is received by a client network, the system comprising:

one or more clients that are interconnected within the client network wherein each client maintains a performance score that is indicative of the clients ability to replicate and distribute the data stream and wherein the client with the highest performance score is designated as a site leader to receive the data stream from the server and wherein the site leader is further configured replicate and re-transmit the data stream to clients to thereby reduce the server bandwidth required to distribute the data stream to each client in the client network.

52. The system of Claim 51, wherein the site leader distributes the data stream to at least one client located in a local subnetwork using a broadcastable transmission mode in which the broadcasted data stream is simultaneous received by each client located in the local subnetwork and wherein the bandwidth consumed when distributing the data stream using the broadcastable transmission mode is substantially the same for one or more clients.

53. The system of Claim 51, wherein the site leader distributes the data stream to at least one client located in a remote subnetwork using a guaranteed delivery mode and wherein the client located in the remotely located subnetwork acts as a subnet leader to replicate and rebroadcast the data stream to at least one client located in the remote subnetwork.

54. A method for moderating streaming data between a content server and a plurality of clients so as to reduce data interruptions; the method comprising:

designating a first leader from the plurality of clients;

transmitting the streaming data from the content server to the first leader wherein the first leader buffers at least a portion of the streaming data and subsequently distributes the streaming data to the plurality of clients;

designating a second leader from the plurality of clients upon receiving notification from the first leader of an impending streaming content distribution interruption whereupon the content server performs a transmission switch to begin transmitting the streaming data to the second leader which buffers at least a portion of the streaming data; and

performing a distribution switch whereupon the first leader coordinates distribution of the streaming data with the second leader and

wherein the buffered streaming data on each leader is used in conjunction with a splicing operation to transparently switch between streaming data from the first leader to streaming data from the second leader.

5 55. The method of Claim 54, wherein the splicing operation utilizes the buffered streaming data of the first and second leaders and identifies a region of streaming content overlap present in each buffer and thereafter performs the distribution switch in the overlapping region to thereby inhibit loss of the integrity of the streaming data.

10 56. The method of Claim 55, wherein during the distribution switch substantially any duplicated data present in the overlap between the first and second buffers is removed to thereby generate substantially continuous streaming data.

15 57. The method of Claim 54, wherein the splicing operation is performed using buffered streaming data before the streaming data has been distributed to the plurality of clients to thereby transition the transmission of streaming data from the first leader to the second leader in a transparent manner.

20 58. A method for moderating streaming content between a client acting as a content distribution source which receives streaming content from a content source and thereafter distributes the streaming content to a plurality of clients so as to reduce data interruptions; the method comprising:

designating a first subnet leader from at least a portion of the plurality of clients located in a different subnet from that occupied by the content distribution source;

25 transmitting the streaming content from the content distribution source to the first subnet leader wherein the first subnet leader buffers at least a portion of the streaming data and subsequently distributes the streaming content to the plurality of clients;

30 designating a second subnet leader from the plurality of clients located in substantially the same subnet as the first subnet leader, which upon receiving notification from the first subnet leader of an impending streaming content distribution interruption, the content distribution source performs a transmission switch to begin transmitting the streaming content

to the second subnet leader which buffers at least a portion of the streaming content; and

5 performing a distribution switch whereupon the first subnet leader coordinates distribution of the streaming content with the second subnet leader and wherein the buffered streaming content on each subnet leader is used in conjunction with a splicing operation to transparently switch between streaming content from the first subnet leader to streaming data from the second subnet leader.

10 59. The method of Claim 58, wherein the splicing operation utilizes the buffered streaming content of the first and second subnet leaders and identifies a region of streaming content overlap present in each buffer and thereafter performs the distribution switch in the overlapping region to thereby preserve the integrity of the streaming data.

15 60. The method of Claim 59, wherein during the distribution switch substantially any duplicated content present in the overlap between the first and second buffers is removed to thereby generate substantially continuous streaming content.

20 61. The method of Claim 58, wherein the splicing operation is performed using buffered streaming content before the streaming content has been distributed to the plurality of clients to thereby transition the transmission of streaming content from the first subnet leader to the second subnet leader in a transparent manner.

25 62. The method of Claim 58, wherein the content distribution source is a content server, the first subnet leader is a first site leader which receives streaming content from the content server and distributes the streaming content to the plurality of clients, and the second subnet leader is a second site leader which will receive streaming content from the content server and distribute the streaming content to the plurality of clients served by the first subnet leader following the distribution switch.

30 63. The method of Claim 58, wherein the content distribution source is a site leader and the first subnet leader is a client located outside of the subnet of the site leader which receives streaming content from the site leader and distributes the streaming content to the plurality of clients, and wherein the second subnet leader

is a second client located outside of the subnet of the site leader and within substantially the same subnet as the first subnet leader which will receive streaming content from the site leader and distribute the streaming content to the plurality of clients served by the first subnet leader following the distribution switch.

5 64. The method of Claim 58, wherein the content distribution source is a content distribution subnet leader and the first subnet leader is a client located outside of the subnet of the content distribution subnet leader which receives streaming content from the content distribution subnet leader and distributes the streaming content to the plurality of clients, and wherein the second subnet leader
10 is a second client located outside of the subnet of the content distribution subnet leader and within substantially the same subnet as the first subnet leader which will receive streaming content from the content distribution subnet leader and distribute the streaming content to the plurality of clients served by the first subnet leader following the distribution switch.

15 65. The method of Claim 58, wherein the content distribution subnet leader is arranged in a hierarchical ordering such that the content distribution subnet leader distributes streaming content to the first and second subnet leaders which are subordinate to the content distribution subnet leader.

20 66. A system for maintaining substantially uninterrupted streaming content, the system comprising:

 a streaming content distribution server which transmits streaming content;

 a first content leader which receives the transmitted streaming content from the streaming content distribution server and buffers at least a
25 portion of the streaming content in a first stream buffer and thereafter distributes the streaming content to at least one client device; and

 a second content leader which receives the transmitted streaming content from the streaming content distribution server and buffers at least a portion of the streaming content in a second stream buffer and wherein at
30 least a portion of buffered streaming content that is commonly contained in both the first stream buffer and the second stream buffer is identified as a stream jump position where a streaming content transition takes place such that the first content leader ceases to distribute the streaming data to the at

least one client device and the second content leader commences with distributing the streaming content to the at least one client device at the stream jump position to provide the at least one client device with substantially uninterrupted distribution of streaming content.

5 67. The method of Claim 66, wherein the streaming data transition takes place upon receiving notification that the first content leader will cease distributing the streaming content to the at least one client device.

68. The method of Claim 66, wherein buffered streaming data provides a means for performing the streaming data transition in a substantially transparent
10 manner in which the client devices receive substantially continuous streaming content distribution.

69. The method of Claim 66, wherein duplicated streaming data contained in the first and second stream buffers is removed to provide substantially continuous streaming content distribution.

15 70. A method for distributing streaming content transmitted by a content provider to generate substantially uninterrupted streaming content distribution, the method comprising:

buffering at least a portion of a first stream of streaming content transmitted by the content provider in a first buffer and buffering at least a
20 portion of a second stream of streaming content transmitted by the content provider in a second buffer wherein the first stream of streaming content is substantially identical to at least a portion of the second stream of streaming content; and

joining the streaming content from the first and second streams at a
25 position where the streams overlap in the first and second buffer to generate a continuous data stream such that the content stream is distributed as a continuous and uninterrupted sequence.

71. The method of Claim 70, wherein during the operation of joining the streaming content from the first and second streams any duplicated streaming
30 content is removed to facilitate generating the continuous data stream.

72. The method of Claim 70, wherein the content provider is a content server and the first and second buffers reside on separate clients that function as site leaders.

73. The method of Claim 70, wherein the content provider is a site leader and the first and second buffers reside on separate clients that function as subnet leaders.

5 74. The method of Claim 70, wherein the content provider is a first subnet leader and the first and second buffers reside on separate clients that function as subordinate subnet leaders to the first subnet leader.

75. A method of switching a source of streaming content from a first content provider to a second content provider, wherein the streaming content is buffered by a content recipient and wherein the streaming content is arranged in a selected
10 sequence, the method comprising:

determining that a first source of streaming content provided by the first content provider is interrupted such that it will no longer provide streaming content to the content recipient;

15 requesting a second source of streaming content from the second content provider, wherein at least a portion of the streaming content provided by the first and second content providers is substantially the same;

receiving the streaming content from the second source of streaming content into a buffer wherein the buffer size is selected to allow simultaneous buffering of at least a portion of the streaming content provided by the first content provider and at least a portion of the streaming
20 content provided by the second content provider; and

removing overlapping portions of the streaming content provided by the first and second content providers and appending the streaming content from the second content provider to the streaming content from the first
25 content provider to generate substantially continuous and uninterrupted sequence of streaming content.

76. A method of distributing streaming content from a content source to a plurality of networked clients wherein each of the plurality of clients includes stream buffering and broadcast capabilities, the method comprising:

30 determining a performance capability for each of the plurality of clients;

selecting a current leader from the plurality of clients based upon the determined performance capabilities;

delivering the streaming content from the content source to the current selected leader; and

5 broadcasting the streaming content from the current selected leader to the remaining plurality of clients so that the current selected leader and the plurality of clients receive the streaming content.

77. The method of Claim 76, wherein performance capabilities are periodically reassessed during the broadcast of the streaming content.

78. The method of Claim 77, further comprising:

10 selecting a new leader based upon the periodic reassessment of performance capability wherein the periodic reassessment determines that one of the clients has a better performance capability than the current selected leader;

delivering the streaming content from the content source to the newly selected leader;

15 ceasing broadcast by the current selected leader and initiating broadcast of the streaming content from the newly selected leader to the remaining plurality of clients.

79. The method of Claim 78, further comprising:

20 performing a stream splicing operation which provides a substantially uninterrupted transition between ceasing broadcast by the current selected leader and the newly selected leader.

80. The method of Claim 78, further comprising:

25 performing a stream splicing operation which provides a substantially transparent transition between ceasing broadcast by the current selected leader and the newly selected leader.

81. The method of Claim 76, wherein determining the performance capabilities for each of the plurality of clients comprises developing a performance score for each of the plurality of clients.

30 82. The method of Claim 81, wherein developing a performance score for each of the plurality of clients comprises dynamically evaluating performance factors selected from the group consisting of processor speed, computational load, bandwidth capacity, latency, availability to decode and distribute streaming content and broadcasting ability.

83. The method of Claim 82, further comprising periodically comparing the performance scores for each of the clients.

84. The method of Claim 83, wherein selecting a new current leader from the plurality of clients comprises comparing performance scores to determine
5 which client has a higher performance score.

85. The method of Claim 82, further comprising periodically broadcasting the performance scores between each of the clients.

86. The method of Claim 78, wherein selecting a new current leader from the plurality of clients comprises each client evaluating the broadcasted
10 performance scores to determine which client has a higher performance score and selecting this client to be the new current leader.

87. The method of Claim 78, wherein the client selected as the new leader broadcasts a leader announcement to the clients following a suppression period for announcements to accommodate transitioning between the current selected leader
15 and the newly selected leader.

88. The method of Claim 76, wherein broadcasting the streaming content from the current selected leader to the plurality of clients comprises broadcasting the streaming content through a subnet using a User Datagram Protocol (UDP).

89. The method of Claim 76, wherein delivering streaming content from the
20 content source to the current selected leader comprises delivering the streaming content using a guaranteed delivery protocol.

90. The method of Claim 76, wherein delivering the streaming content from the content source to the current selected leader comprises delivering the streaming content from a site leader to the current selected leader of a subnet
25 using a Transmission Control Protocol/Internet Protocol (TCP/IP).

91. The method of Claim 76, further comprising adding a new client to the plurality of clients.

92. The method of Claim 76, further comprising determining whether the performance capability of the new client exceeds the current selected leader.

30 93. The method of Claim 92, further comprising:
delivering the streaming content from the content source to the new client when the new client is determined to have a better performance

capability than the current selected leader such that the new client comprises a newly selected leader;

ceasing broadcast by the current selected leader;

initiating broadcast of the streaming content from the newly selected leader to the remaining plurality of clients.

94.A method for distributing streaming content to a plurality of clients to maintain consistency in data throughput, the method comprising the steps of:

(a) conducting a performance evaluation for each of the plurality of clients to identify clients with acceptable performance characteristics;

(b) designating the client with the highest performance as a site leader;

(c) transmitting the streaming content from a stream server to the site leader;

(d) broadcasting the streaming content from the site leader to the remaining plurality of clients;

(e) periodically re-evaluating performance for the plurality of clients to identify changes in client performance characteristics; and

(f) repeating steps (a)-(e) to thereby maintain service quality in the distribution streaming content.

95.A method of determining a leader among one or more clients in a subnet, the method comprising:

assuming a client is the subnet leader unless another client is better qualified to be the leader;

assessing leadership qualities of the clients at selected intervals; and providing a transition period that allows leadership to be transitioned from one client to another.

96.A method of determining a leader in a subnet having a plurality of clients wherein the leader receives a data stream from a source and distributes the data stream to clients, the method comprising:

assuming each client to be the leader unless another client is better qualified to be the leader based upon a performance assessment to thereby

establish a current leader that is the most qualified of the plurality of clients based upon the performance assessment;

announces leadership by the current leader recognized by non-leader clients which listen to the current leader's announcement

5 periodically performing a performance assessment for each of the non-leader clients and comparing the performance assessment to the announcing leader's performance assessment to determine whether at least one of the non-leader clients is better qualified to assume leadership within the subnet; and

10 providing a transition state that allows the non-leader client with the highest performance assessment greater than the current leader to assume leadership of the subnet as a new leader wherein the current leader becomes a non-leader client.

15 97. A method of determining a leader among one or more members of a subnet, the method comprising:

allowing each member to be the leader unless another member is better qualified to be the leader;

assessing leadership qualities of the members at selected instances; and

20 providing a transition period that allows transition of leadership from one member to another.

98. The method of Claim 97, wherein the leadership quality of the member comprises a performance score indicative of the member's ability to receive and distribute the data stream.

25 99. The method of Claim 98, wherein the performance score is based on the member's processor speed, computational load, and bandwidth capacity.

100. The method of Claim 99, wherein assessing leadership qualities of the members occurs periodically.

30 101. The method of Claim 100, wherein the assessment comprises a periodic announcement of the leader's score to other members.

102. The method of Claim 101, wherein assessing leadership qualities of the members occurs whenever a member enters or exits the subnet.

103. The method of Claim 102, wherein any sole member of the subnet is the subnet leader.

104. The method of Claim 103, wherein a new entering member enters a transition state that lasts for a transition period during which it listens to the existing leader's announcement.

105. The method of Claim 104, wherein the length of transition period is greater than the period between the leader announcements.

106. The method of Claim 105, wherein the entering member assumes leadership from the existing leader if its performance score is greater than that of the existing leader and wherein the new leader enters an announce state during which it announces its performance score to the subnet.

107. The method of Claim 106, wherein the entering member does not become a leader and wherein the entering member becomes a non-leader member and enters a listen state during which it listens to the announcements of the existing leader and compares the existing leader's performance score to the entering member's performance score.

108. The method of Claim 107, wherein if the non-leader member has a higher performance score than that of the leader, the non-leader leaves the listen state and enters the transition state prior to assuming leadership from the existing leader.

109. The method of Claim 108, wherein the existing leader relinquishes leadership and transitions from an announce state to the listen state

110. The method of Claim 108, wherein the subnet leader receives the data stream from an upstream source and further transmits it to other subnet leaders.

111. The method of Claim 108, wherein the subnet leader receives the data stream from an upstream source and distributes it to members within the same subnet.

112. The method of Claim 111, wherein the subnet leader broadcasts the data stream to the non-leader members in the subnet.

113. A system for determining a leader in a subnet wherein the subnet comprises one or more clients that are capable of acting as the subnet leader, the system comprising:

5 a process that runs on each of the clients wherein the process determines a leadership quality for the client and wherein the process periodically compares the leadership quality of the client to that of other clients within the subnet such that the client with the best leadership quality assumes leadership.

10 114. The system of Claim 113, wherein the process is a software application that runs on each of the subnet members.

115. The system of Claim 113, wherein leadership quality is determined based on a subnet member's characteristics selected from the group consisting of: processor speed, computational load, and bandwidth capacity.

15 116. The system of Claim 113, wherein the subnet member is a personal computer.

117. A method of distributing steaming content from a content source to a plurality of networked clients, wherein the plurality of networked clients are logically divided into a plurality of subnets, the method comprising:

20 determining a performance capability for each of the plurality of clients;

 selecting, based upon the performance capability of each of the plurality of clients, a subnet leader for each of the plurality of subnets;

 selecting, based upon the performance capability of each of the plurality of subnet leaders, a site leader;

25 transmitting streaming content from the content source to the selected site leader;

 transmitting the streaming content from the selected site leader to the plurality of subnet leaders; and

30 broadcasting the streaming content from the selected site leader and the plurality of subnet leaders to the clients within the plurality of subnets.

118. The method of Claim 117, further comprising periodically reassessing the performance capabilities of each of the plurality of clients and redetermining

the plurality of subnet leaders and the plurality of site leaders based upon the reassessed performance capabilities.

119. The method of Claim 118, further comprising adding an additional subnet of additional clients to the plurality of networked clients.

5 120. The method of Claim 119, further comprising:
 determining a performance capability to each of the plurality of additional clients in the additional subnet;

 selecting, based upon the performance capabilities of each of the plurality of additional clients an additional subnet leader for the additional
10 subnet;

 transmitting the streaming content from the selected site leader to the additional subnet leader; and

 broadcasting the streaming content from the additional subnet leader to the additional clients within the additional subnet.

15 121. The method of Claim 120, further comprising assessing the performance of the transmission of the streaming content from the selected site leader to the plurality of subnet leaders.

 122. The method of Claim 120, wherein transmitting the streaming content from the selected site leader to the additional subnet leader comprising
20 transmitting the streaming content from the selected site leader directly to the additional subnet leader when the performance assessment of the transmission of the streaming content from the selected site leader to the plurality of subnet leaders satisfies a pre-selected threshold.

 123. The method of Claim 121, wherein the pre-selected threshold
25 comprises a fan-out limit that is defined to be a specific number of subnet leaders.

 124. The method of Claim 121, wherein the pre-selected threshold comprises a fan-out limit that is dynamically determined based upon assessed performance characteristics of the transmission of the streaming content.

 125. The method of Claim 121, wherein transmitting the streaming content
30 from the selected site leader to the additional subnet leader comprises transmitting the content from the selected site leader to a first subnet leader and then re-transmitting the streaming content from the first subnet leader to the additional

subnet leader when the assessment of transmission of the streaming content does not satisfy a selected threshold.

126. The method of Claim 125, wherein determining a performance capability for each of the plurality of clients comprises developing a performance score of each of the plurality of clients and each of the plurality of additional clients.

127. The method of Claim 126, wherein developing a performance score for each of the plurality of clients and plurality of additional clients comprises dynamically evaluating performance factors selected from the group consisting of processor speed, computational load, bandwidth capacity, latency, availability to decode and distribute streaming content and broadcasting ability.

128. The method of claim 126, further comprising dynamically ranking each of the clients and additional clients based on their performance score and wherein periodically redetermining the plurality of subnet leaders and the selected site leader occurs upon changes in the dynamic rankings of each of the clients and additional clients.

129. The method of Claim 126, wherein transmitting the streaming content from the selected site leader to the additional subnet leader comprises transmitting via the first subnet leader comprises selecting the first subnet leader as the leader having the highest performance score.

130. The method of Claim 129, wherein periodically redetermining the plurality of subnet leaders and the selected site leader comprises assigning the additional subnet leader as the site leader when the additional subnet leader has the highest performance score.

131. The method of Claim 130, wherein transmitting the streaming content from the selected site leader to one of the subnet leaders comprises transmitting the streaming content to the subnet leader via the additional subnet leader when the additional subnet leader is determined to have a performance score higher than the subnet leader and the transmission assessment indicates that the transmission performance does not satisfy the pre-selected threshold.

132. The method of Claim 117, wherein transmitting streaming content from the content source to the selected site leader comprises transmitting the streaming content via a public network wherein the streaming content is wrapped in a protocol supported by the public network.

133. The method of Claim 117, wherein transmitting the streaming content from the content source comprises transmitting packetized multiplexed channels of streaming content.

5 134. The method of Claim 117, wherein transmitting streaming content from the content source to the selected site leader comprises transmitting the streaming content via the Internet wherein the streaming content is wrapped in an HTTP protocol.

10 135. The method of Claim 117, further comprising the selected site leader unwrapping and demultiplexing the packetized streaming content into a plurality of channels of broadcast enabled streaming content.

136. The method of Claim 117, wherein transmitting the streaming content from the selected site leader to the plurality of subnet leaders comprises transmitting the plurality of channels of broadcast enabled streaming content using a guaranteed delivery protocol.

15 137. The method of Claim 117, wherein transmitting the streaming content from the site leader to the subnet leader comprises delivering the streaming content from the selected site leader to the plurality of subnet leaders using a Transmission Control Protocol/Internet Protocol (TCP/IP).

20 138. The method of Claim 117, wherein transmitting the streaming content from the plurality of subnet leaders to the clients within the plurality of subnets comprises broadcasting the streaming content through a subnet using a User Datagram Protocol (UDP).

139. A method for improving distribution of streaming content between a content server and a plurality of clients, the method comprising:

- 25 (a) identifying a data stream distribution capacity for each of the plurality of clients;
- (b) identifying a performance score for each of the plurality of clients indicative of the relative efficiency with which each client can broadcast the streaming content to other clients;
- 30 (c) arranging the clients in a hierarchical subnet ordering wherein the client with the highest relative efficiency is designated as a primary site leader which broadcasts streaming content to one or more subordinate clients having lesser performance scores and

wherein the number of subordinate clients does not exceed the data stream distribution capacity of the primary site leader; and if additional clients remain, proceeding to act (d)

5 (d) designating the subordinate client with the highest relative efficiency as a subordinate site leader and arranging the remaining clients in the hierarchical subnet ordering wherein the number of subordinate clients under the subordinate site leader does not exceed the data stream distribution capacity of the subordinate site leader; and if additional clients remain, repeating
10 act (d) until all clients have been arranged in the hierarchical subnet ordering.

140. A method of distributing a data stream to one or more clients in a client network wherein the client network receives the data stream from a server, the method comprising:

15 (a) assessing distribution capabilities of the clients and assigning scores to the clients accordingly;
(b) selecting a highest score client to be a site leader that receive the data stream from the server and distribute the data stream therefrom;
(c) if necessary, selecting a first group of clients to establish client-
20 to-client fan-out connections to the site leader wherein the number of clients in the first group is determined by the site leader's distribution capability;
(d) if necessary, forming fan out distribution of the data stream from one or more clients of the first group to remaining clients; and
(e) repeating acts (a) to (d) as needed.

25 141. The method of Claim 140, wherein assessing distribution capabilities of the clients comprises requesting and receiving a list of clients in the client network.

142. The method of Claim 140, wherein selecting the first group of clients comprises selecting the clients having the highest scores aside from the site
30 leader.

143. The method of Claim 142, wherein remaining clients that fan-out from the first group of clients have lower scores wherein a hierarchy of fan-out

distribution formed below the first group is based on the scores of the clients wherein higher score clients are closer to the first group.

144. The method of Claim 140, wherein the clients in the fan-out distribution are subnet leaders wherein the subnet leader is a member of a subnet most qualified to represent the subnet wherein the subnet leader receives and distributes the data stream to members of the subnet.

145. The method of Claim 144, wherein the fan-out connections between the site leader and the subnet leaders, and the fan-out connection hierarchy therebelow, comprise client-to-client connections.

146. The method of Claim 145, wherein the client-to-client connection utilizes Transmission Control Protocol (TCP) to send the data stream.

147. The method of Claim 140, wherein the subnet leader broadcasts the data stream to members of the subnet.

148. The method of Claim 147, wherein the broadcast utilizes User Datagram Protocol (UDP).

149. The method of Claim 140, wherein the subnet leader forms the client-to-client connection to a subnet member that is not configured to receive the broadcast of the data stream.

150. A method of organizing a distribution of a data stream within a client network having a plurality subnets wherein each subnet is represented by a subnet leader and wherein the data stream is sent from a server to the client network, the method performed by a selected subnet leader comprises:

requesting a list of possible connection sources from the server or a current site leader that represents the most capable subnet leader for receiving the data stream from the server and distributing the data stream therefrom wherein the connection sources comprise subnet leaders that are either already receiving the data stream or requesting the data stream;

receiving the list of connection sources;

analyzing the connection sources wherein a score is assigned to each source on the list and wherein the score is based on the ability of the source to receive and distribute the data stream;

selecting the highest score source to be a new site leader such that the new site leader now receives the data stream from the server;

selecting a first group of subnet leaders to receive replicated data stream from the new site leader via a fan-out connection wherein each subnet leader of the first group forms a client-to-client connection to the site leader;

5 distributing the data stream from each of the subnet leader of the first group to its corresponding subnet members; and

if necessary, further distributing the data stream from selected subnet leader of the first group to remaining subnet leaders.

10 151. A system for transmitting streaming content via a non-multicast supported public network, the system comprising:

 a streaming content source that receives a plurality of streams of streaming content wherein the streaming content source multiplexes the plurality of streams of streaming content into a formatted data stream and wherein the streaming content source wraps the formatted data stream into
15 a protocol supported by the public network and induces transmission of the wrapped formatted data stream via the non-multicast supported public network; and

 a recipient network that includes a plurality of client devices, wherein the recipient network demultiplexes and unwraps the formatted data stream,
20 wherein the recipient network transmits the unwrapped data stream to at least one of the client devices so as to broadcast the streaming content.

 152. The system of Claim 151, wherein the streaming content source includes a tunnel server that encodes and wraps the formatted data stream into a format that permits transmission of the formatted data stream through the non-
25 multicast supported public network.

 153. The system of Claim 152, wherein the tunnel server attaches a tunnel header to the formatted data stream to permit transmission of the formatted data stream through the Internet.

 154. The system of Claim 153, wherein the tunnel header comprises an
30 HTTP protocol header to the formatted data stream to permit transmission of the formatted data stream through the Internet.

155. The system of Claim 152, wherein the streaming content source includes at least one stream encoder and at least one media server that digitally encodes raw content data.

5 156. The system of Claim 155, wherein the at least one stream encoder and media server provides data to the tunnel server in either a multicast transmission format, a broadcast format or a guaranteed delivery format.

157. The system of Claim 151, wherein the recipient network includes a first client device that is designated a site leader and a plurality of client devices that are organized into subnets and wherein the site leader unwraps the data stream and transmits the data stream to the plurality of subnets for subsequent broadcast to the plurality of client devices in the plurality of subnets.

158. The system of Claim 157, wherein the plurality of subnets are organized to include one client device that is designated a subnet leader.

159. The system of Claim 158, wherein the site leader unwraps the formatted data stream by removing the tunnel header.

160. The system of Claim 159, wherein the site leader transmits the unwrapped formatted data stream to the plurality of subnet leaders using a guaranteed delivery protocol.

20 161. The system of Claim 160, wherein the site leader attaches a guaranteed delivery header to the unwrapped formatted data stream for transmission to the subnet leaders.

162. The system of Claim 161, wherein the guaranteed delivery header comprises a TCP/IP communications protocol header.

25 163. The system of Claim 162, wherein each of the plurality of subnet leaders perform error correction associated with the guaranteed delivery protocol and demultiplex the formatted data stream.

164. The system of Claim 163, wherein each of the plurality of subnet leaders broadcast the demultiplexed formatted data streams on a plurality of channels in a broadcast format to at least one of the client devices in the associated subnet.

165. The system of Claim 164, wherein the plurality of subnet leaders attach a UDP header to the demultiplexed formatted data streams.

166. A system for transmitting streaming content via a non-multicast supported public network, the system comprising:

5 a streaming content source that transmits a plurality of streams of streaming content via a broadcast or multicast satellite network wherein the streaming content source multiplexes the plurality of streams of streaming content into a formatted data stream and wherein the streaming content source wraps the formatted data stream into a protocol supported by the public network and induces transmission of the wrapped single formatted data stream via the non-multicast supported public network; and

10 a recipient network that includes a plurality of client devices, wherein the recipient network demultiplexes and unwraps the formatted data stream, wherein the recipient network transmits the unwrapped data stream to at least one of the client devices so as to broadcast the streaming content.

15 167. The system of Claim 166, wherein the streaming content source includes a tunnel server that encodes and wraps the formatted data stream into a format that permits transmission of the formatted data stream through the non-multicast supported public network.

20 168. The system of Claim 167, wherein the tunnel server attaches a tunnel header to the formatted data stream to permit transmission of the formatted data stream through the Internet.

169. The system of Claim 168, wherein the tunnel header comprises an HTTP protocol header to the formatted data stream to permit transmission of the formatted data stream through the Internet.

25 170. The system of Claim 167, wherein the streaming content source includes at least one stream encoder and at least one media server that digitally encodes raw content data.

30 171. The system of Claim 170, wherein the broadcast or multicast satellite network includes an uplink dish, a satellite, a downlink dish and a network ready satellite decoder wherein the uplink dish receives the digitally encoded raw content data from the at least one steam encoder and at least one media server and wherein the network ready satellite decoder provides the digitally encoded raw content data to the tunnel server which is located remotely from the at least one digital encoder and the at least one media server.

172. The system of Claim 171, wherein the tunnel server is integrated into the network ready satellite decoder.

173. The system of Claim 170, wherein the at least one stream encoder and media server provides data to the tunnel server in either a multicast
5 transmission format, a broadcast format or a guaranteed delivery format.

174. The system of Claim 166, wherein the recipient network includes a first client device that is designated a site leader and a plurality of client devices that are organized into subnets and wherein the site leader unwraps the data stream and transmits the data stream to the plurality of subnets for subsequent
10 broadcast to the plurality of client devices in the plurality of subnets.

175. The system of Claim 174, wherein the plurality of subnets are organized to include one client device that is designated a subnet leader.

176. The system of Claim 175, wherein the site leader unwraps the formatted data stream by removing the tunnel header.

15 177. The system of Claim 176, wherein the site leader transmits the unwrapped formatted data stream to the plurality of subnet leaders using a guaranteed delivery protocol.

178. The system of Claim 177, wherein the site leader attaches a guaranteed delivery header to the unwrapped formatted data stream for
20 transmission to the subnet leaders.

179. The system of Claim 178, wherein the guaranteed delivery header comprises a TCP/IP communications protocol header.

180. The system of Claim 177, wherein each of the plurality of subnet leaders perform error correction associated with the guaranteed delivery protocol
25 and demultiplex the formatted data stream.

181. The system of Claim 180, wherein each of the plurality of subnet leaders broadcast the demultiplexed formatted data streams on a plurality of channels in a broadcast format to at least one of the client devices in the associated subnet.

30 182. The system of Claim 181, wherein the plurality of subnet leaders attach a UDP header to the demultiplexed formatted data streams.

FIGURE 1

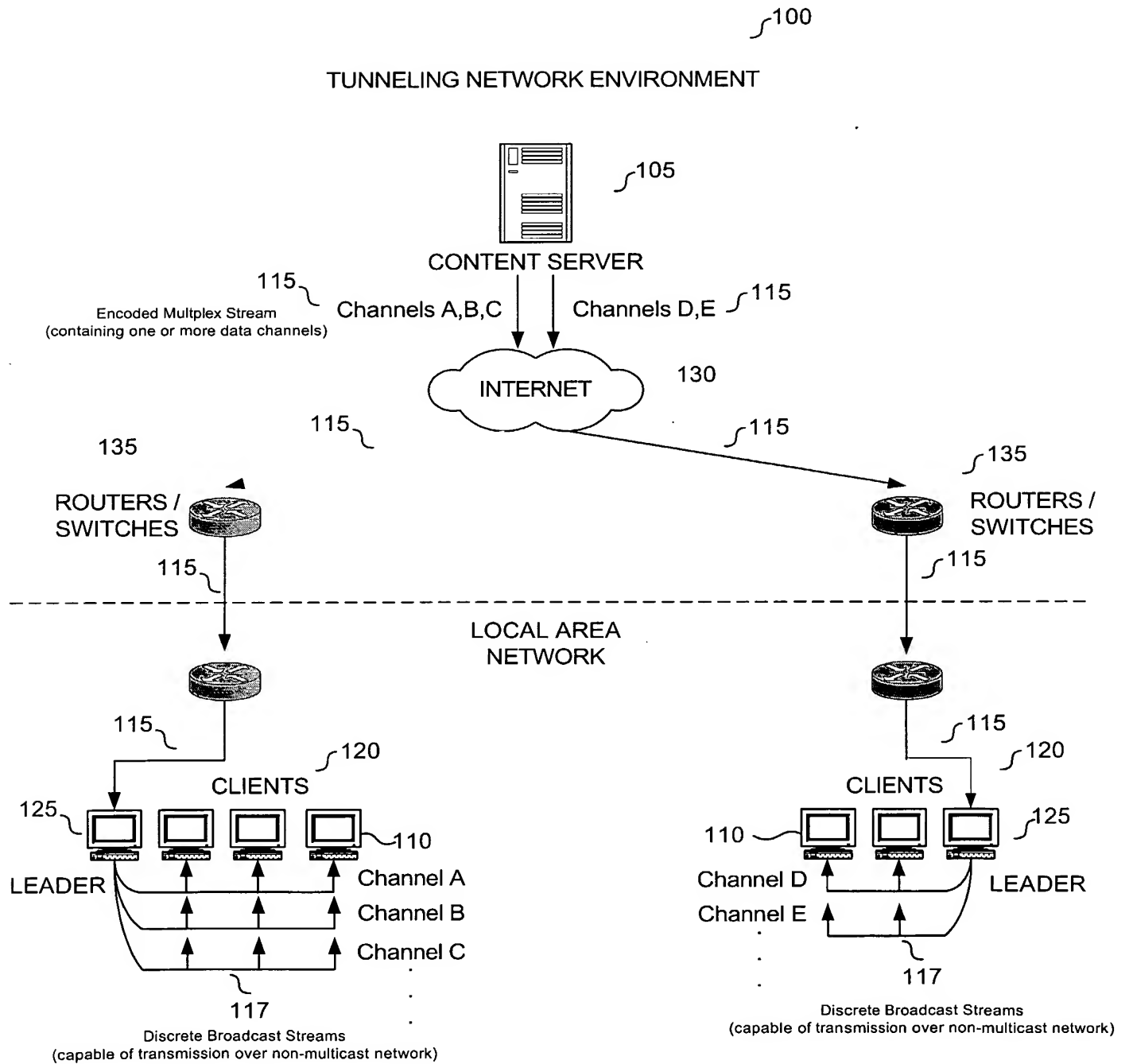


FIGURE 2A

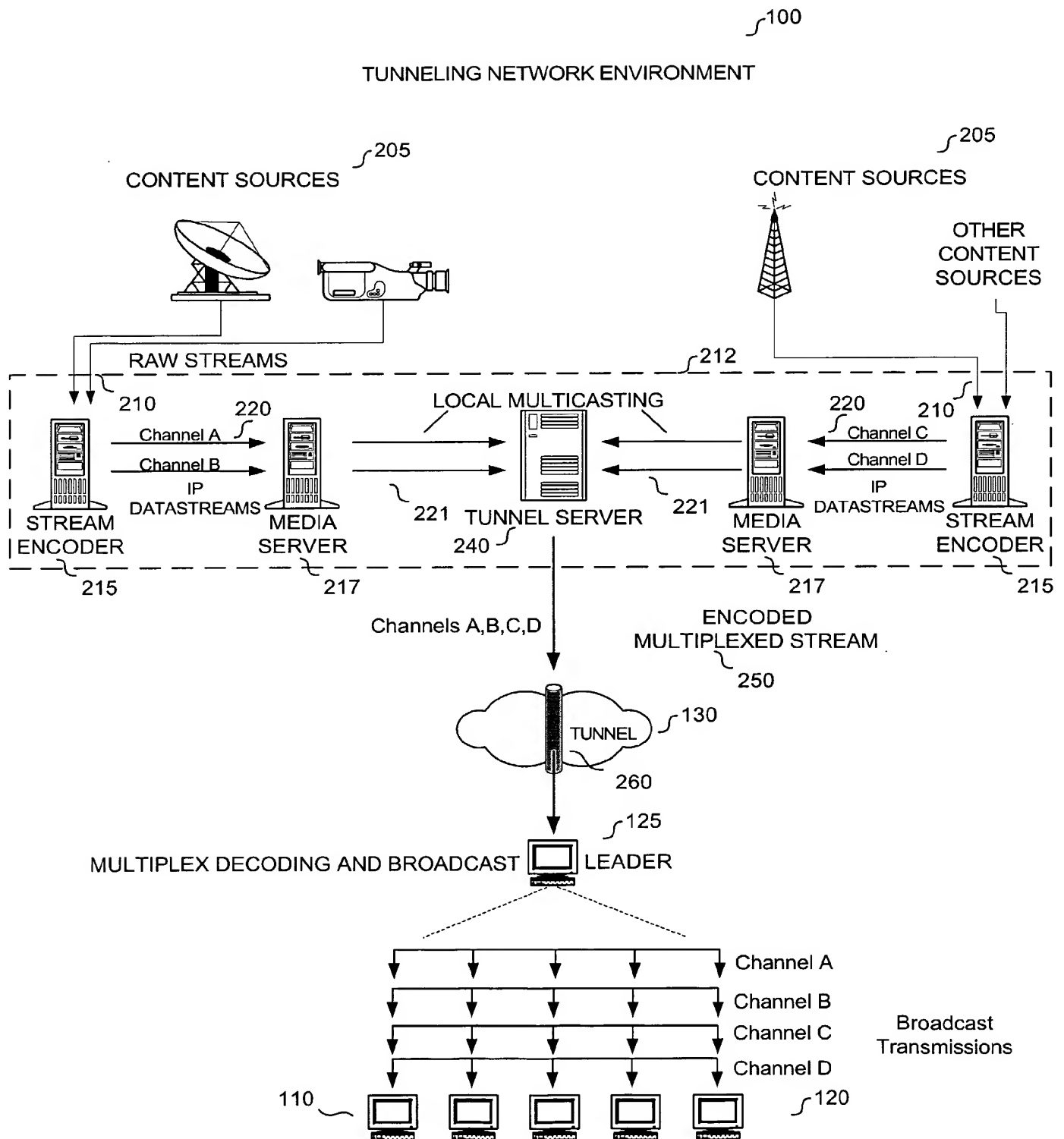


FIGURE 2B

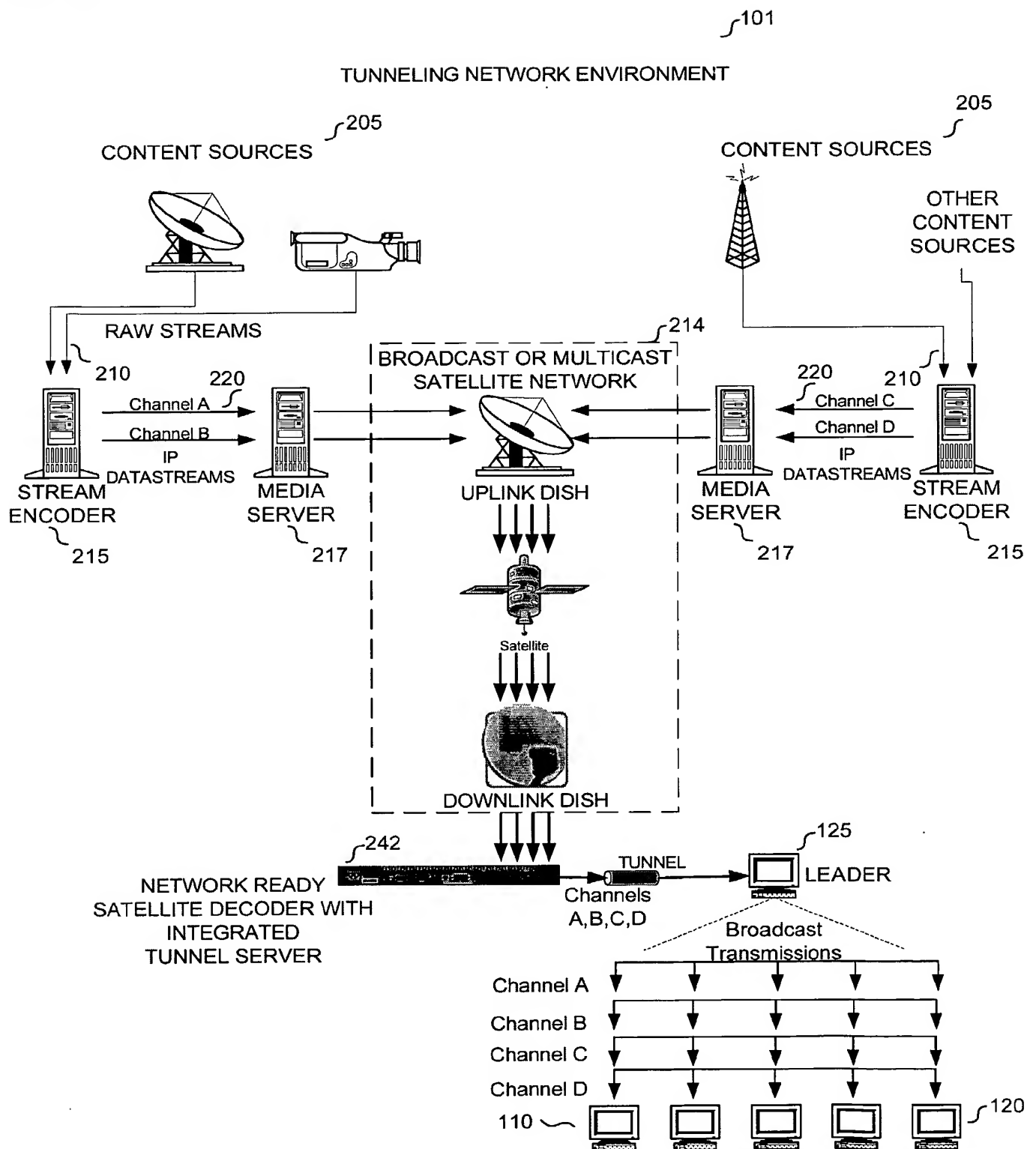


FIGURE 2C

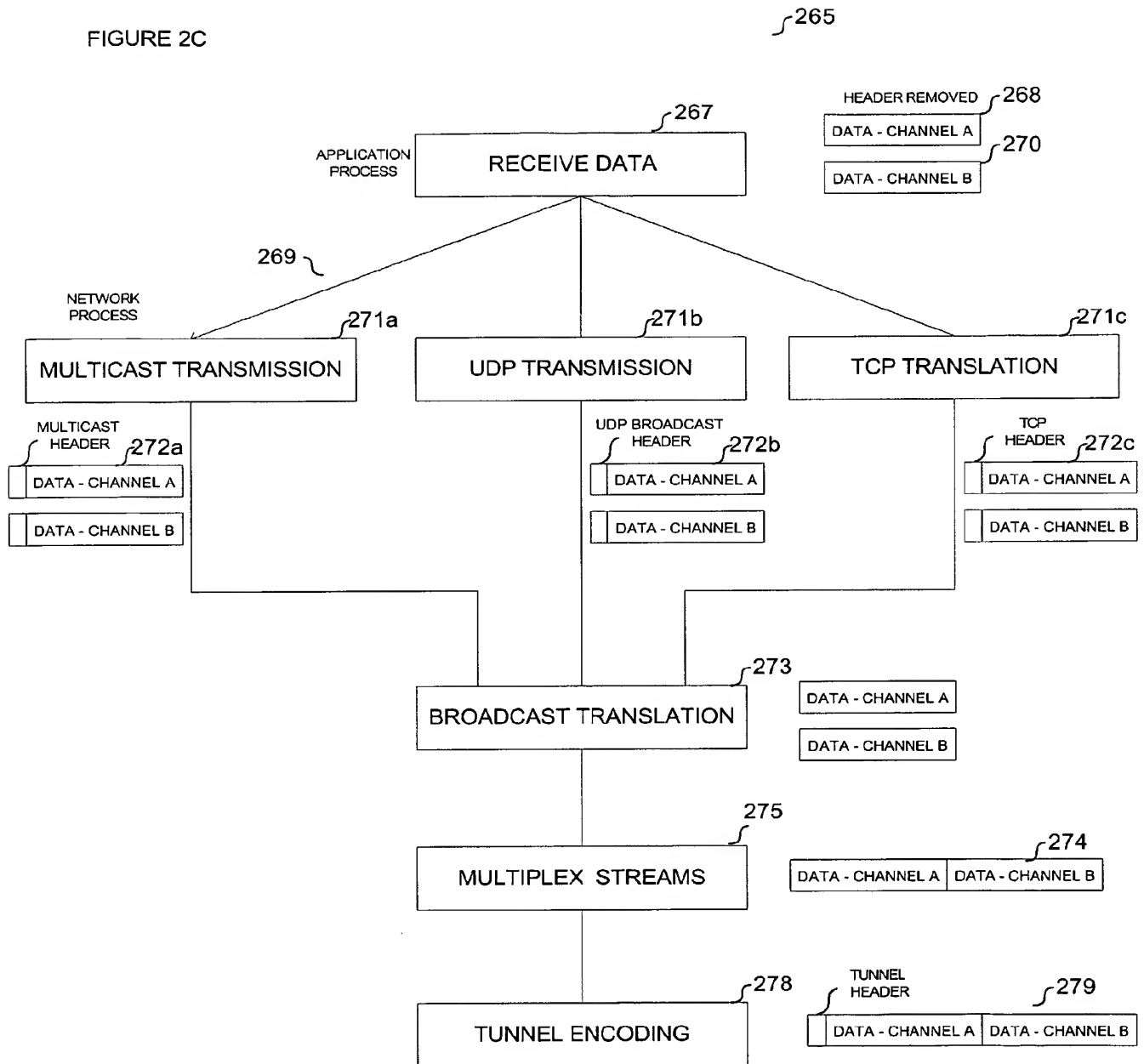


FIGURE 2D

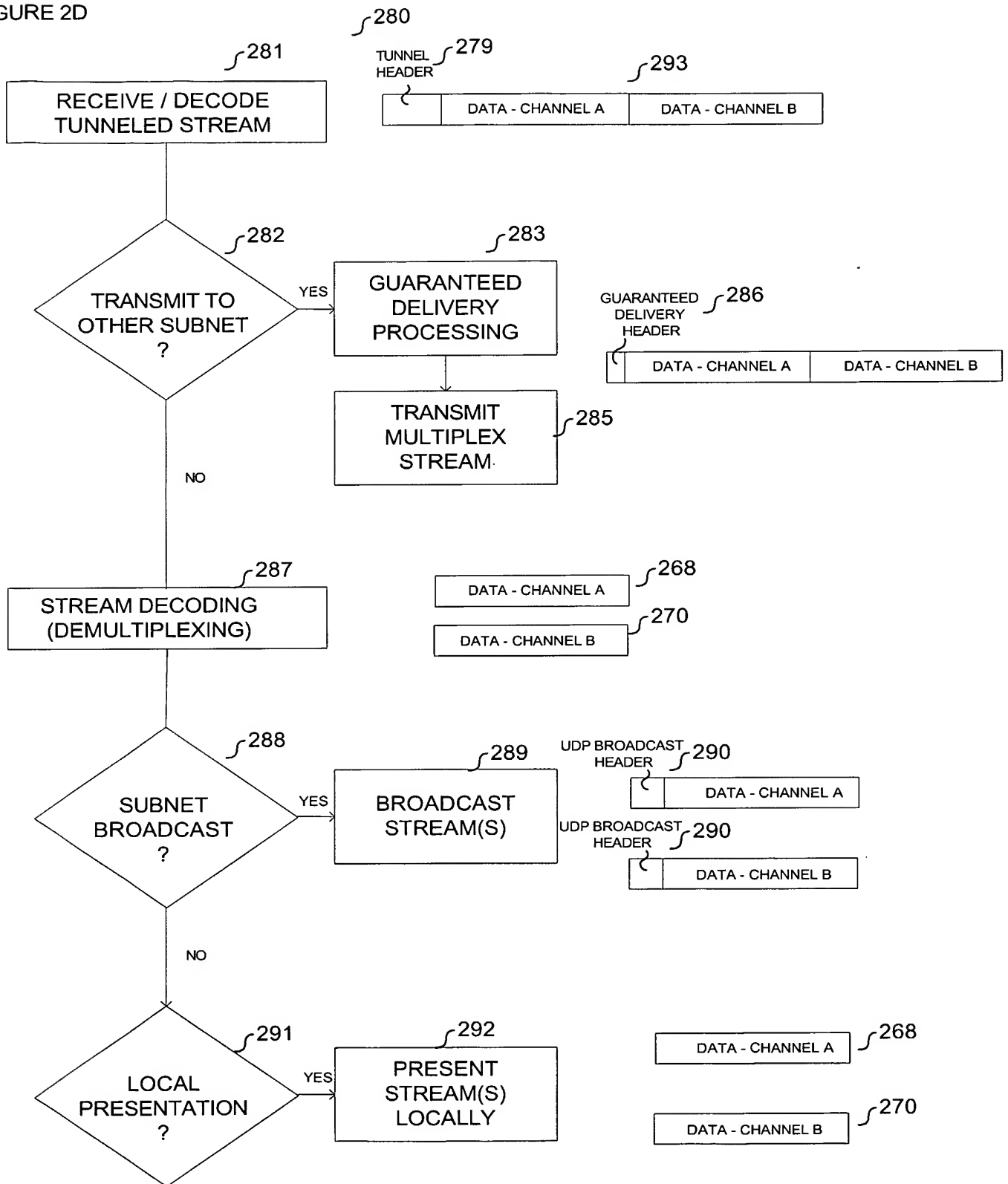


FIGURE 3A

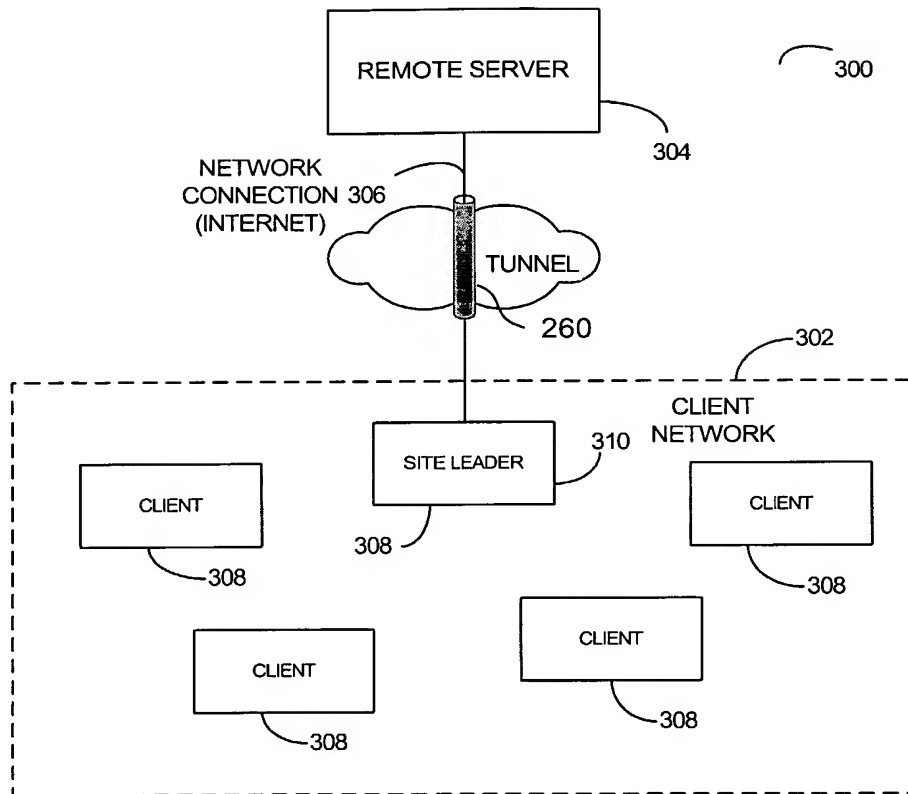


FIGURE 3B

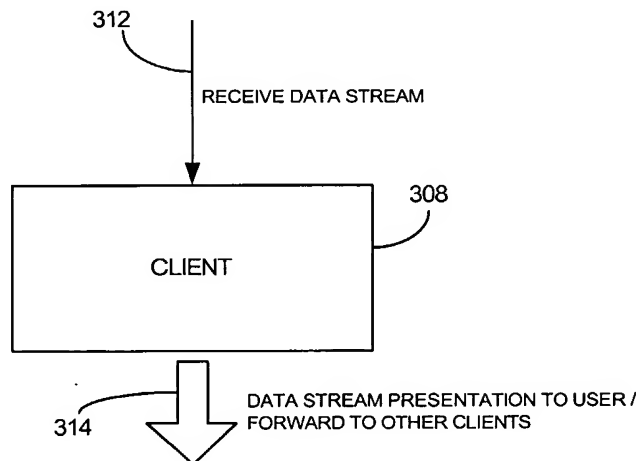


FIGURE 4

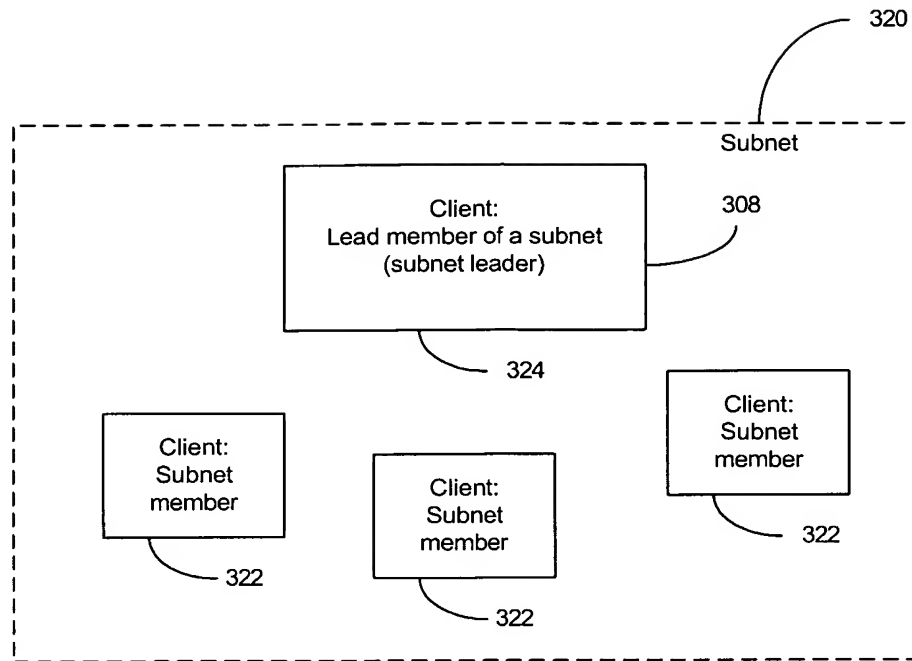


FIGURE 5A

DATA STREAM DISTRIBUTION METHODOLOGY

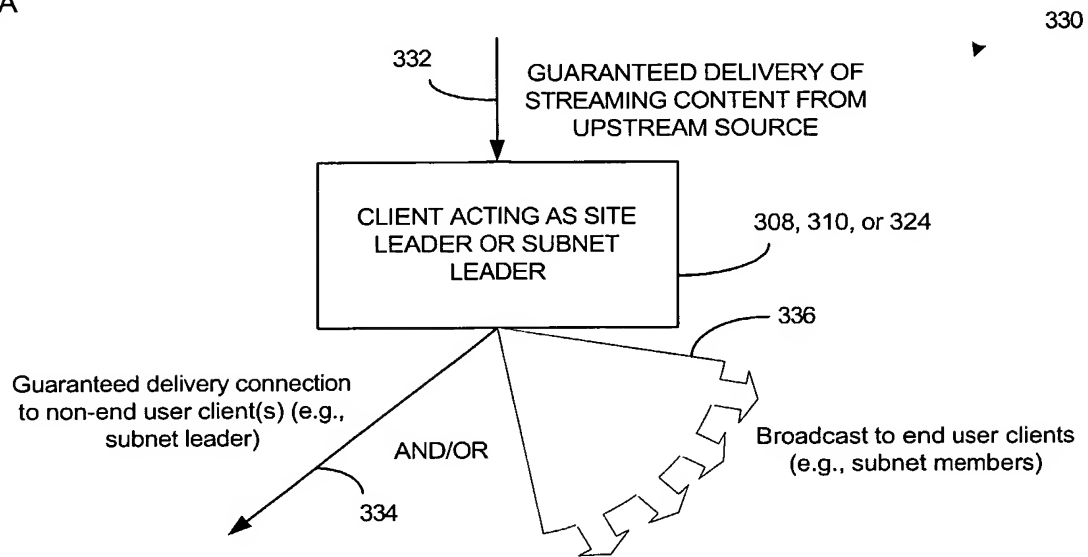


FIGURE 5B

Preferred data stream protocols

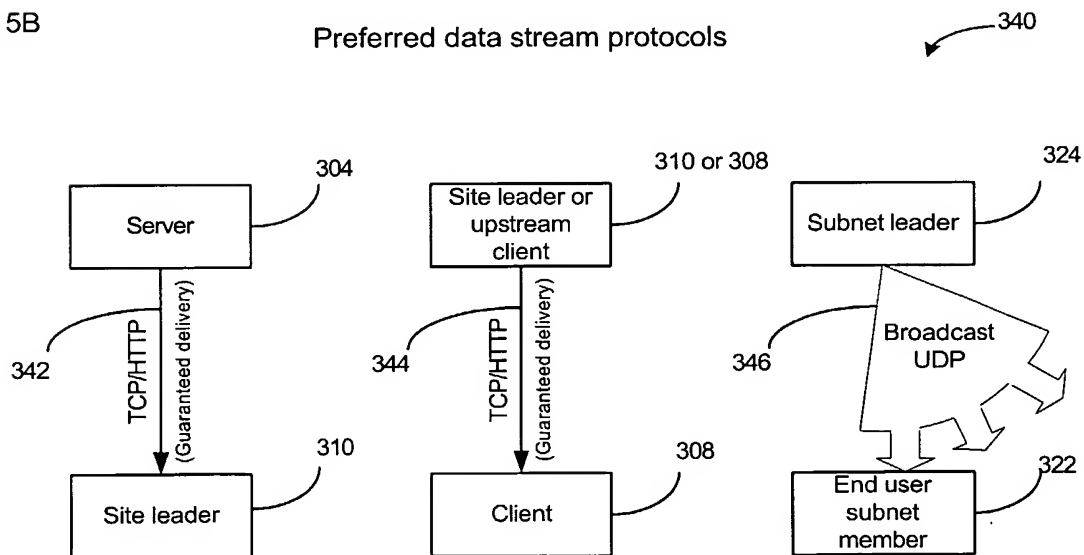
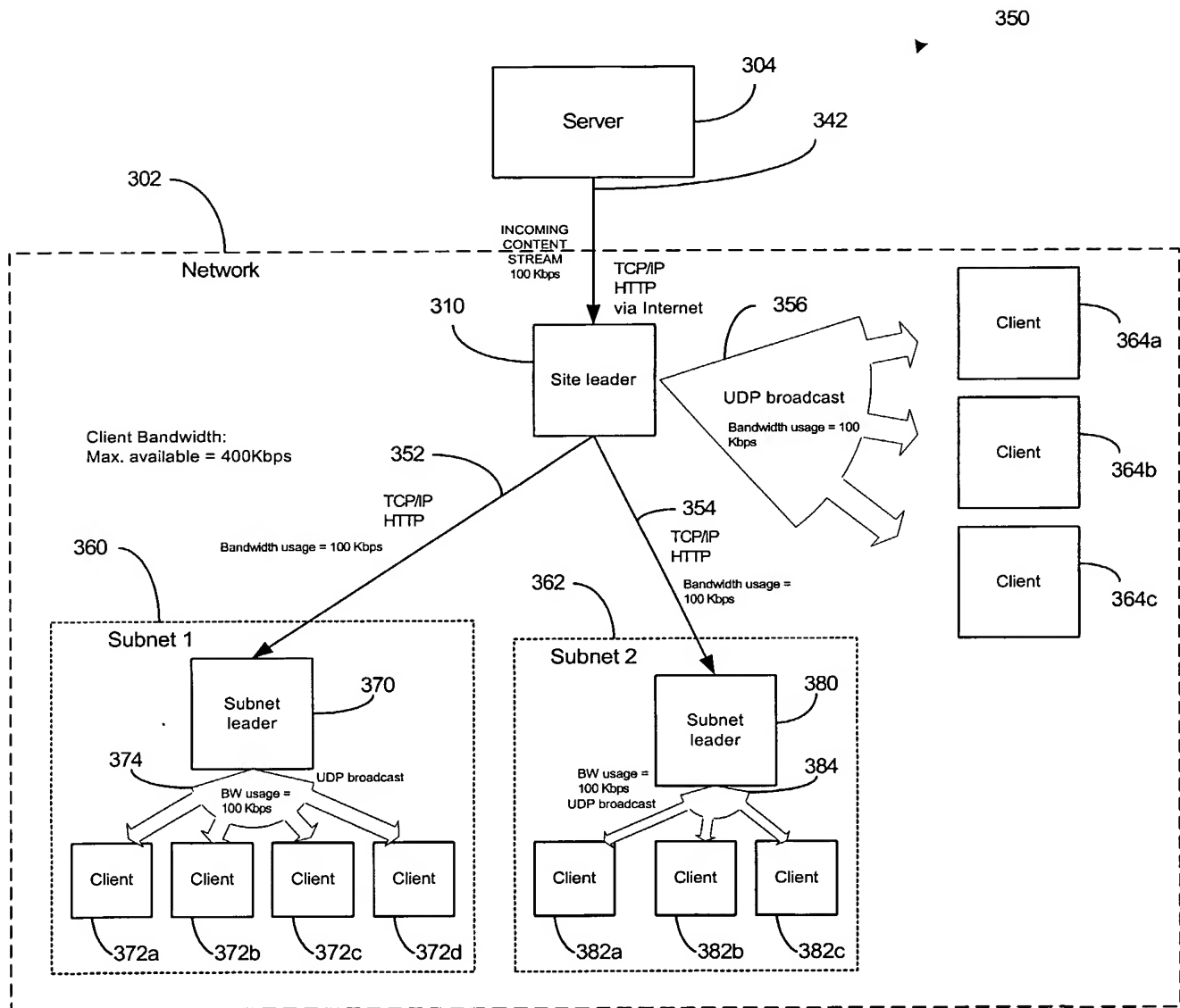


FIGURE 6A

Exemplary multicasting of streaming data



SUBNET STREAM BROADCASTING

FIGURE 6B

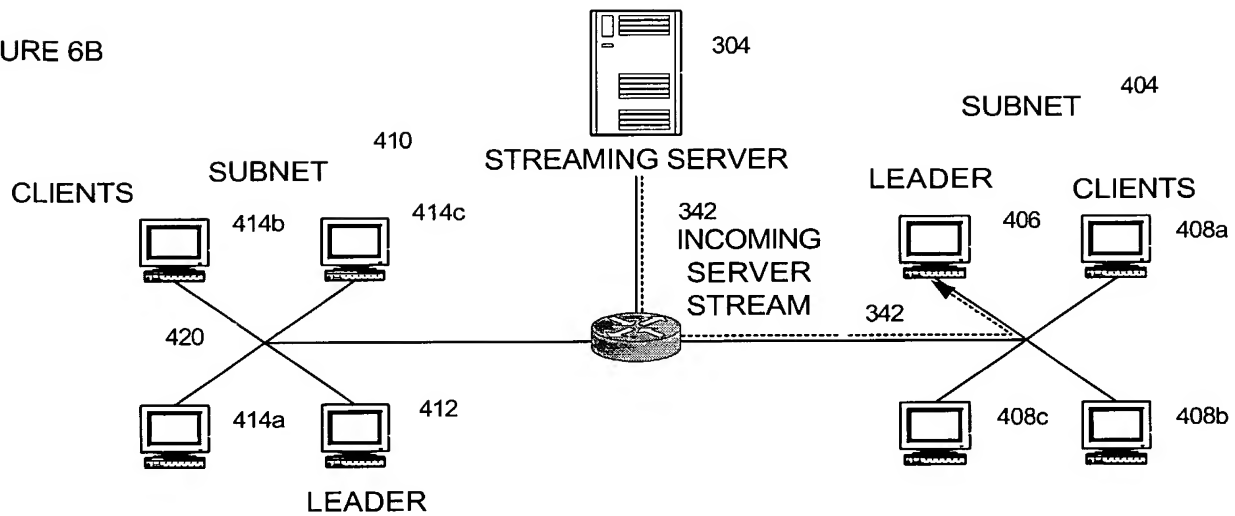
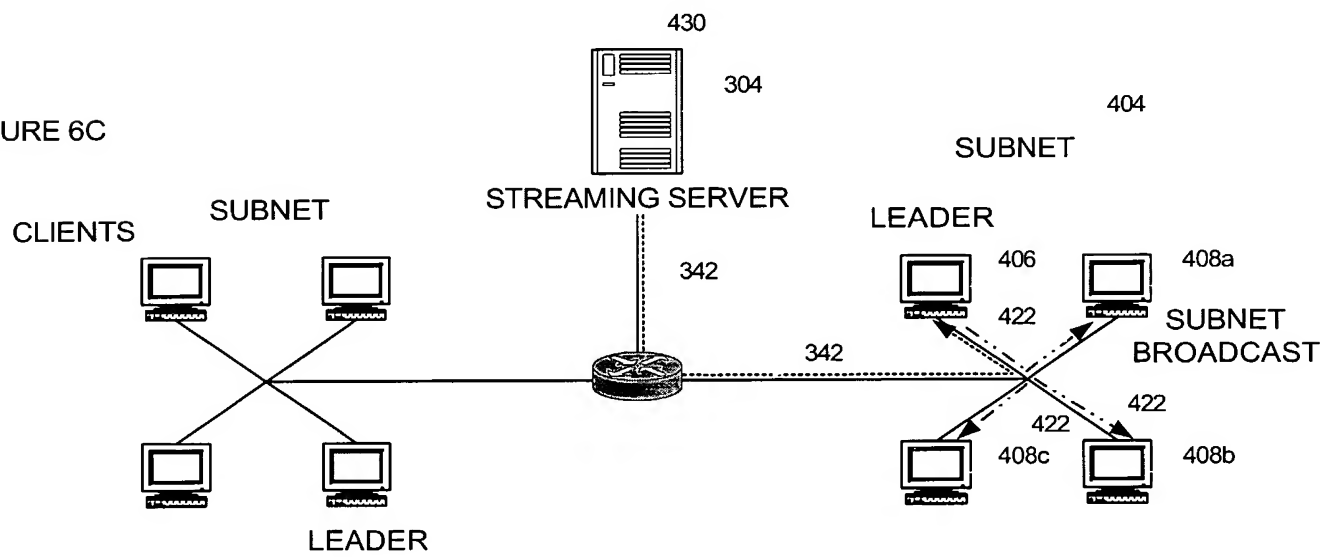


FIGURE 6C



SUBNET STREAM BROADCASTING

FIGURE 6D

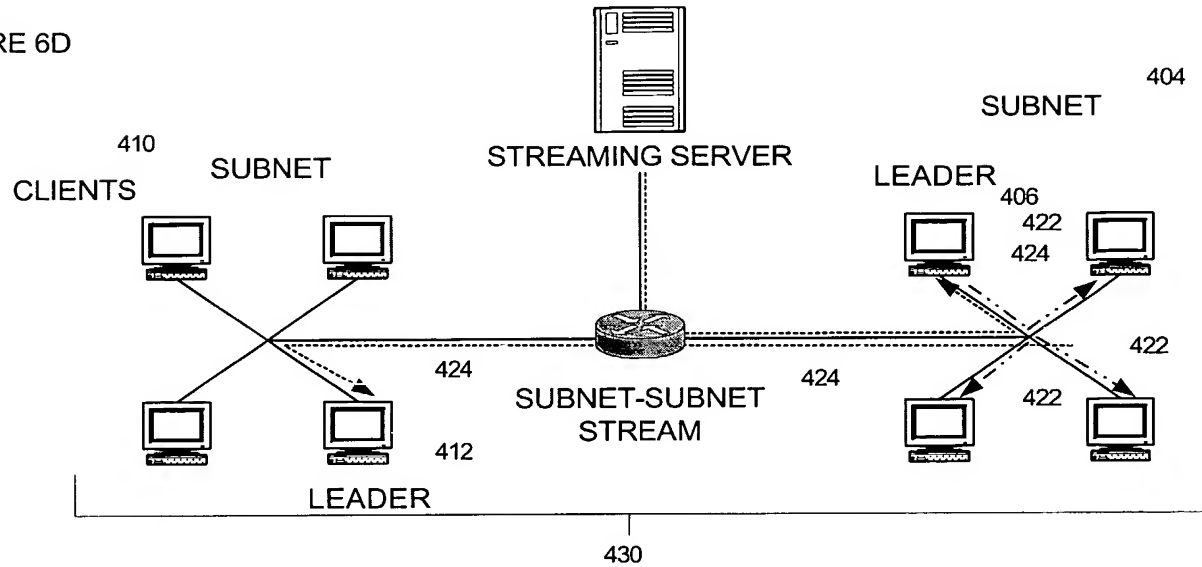


FIGURE 6E

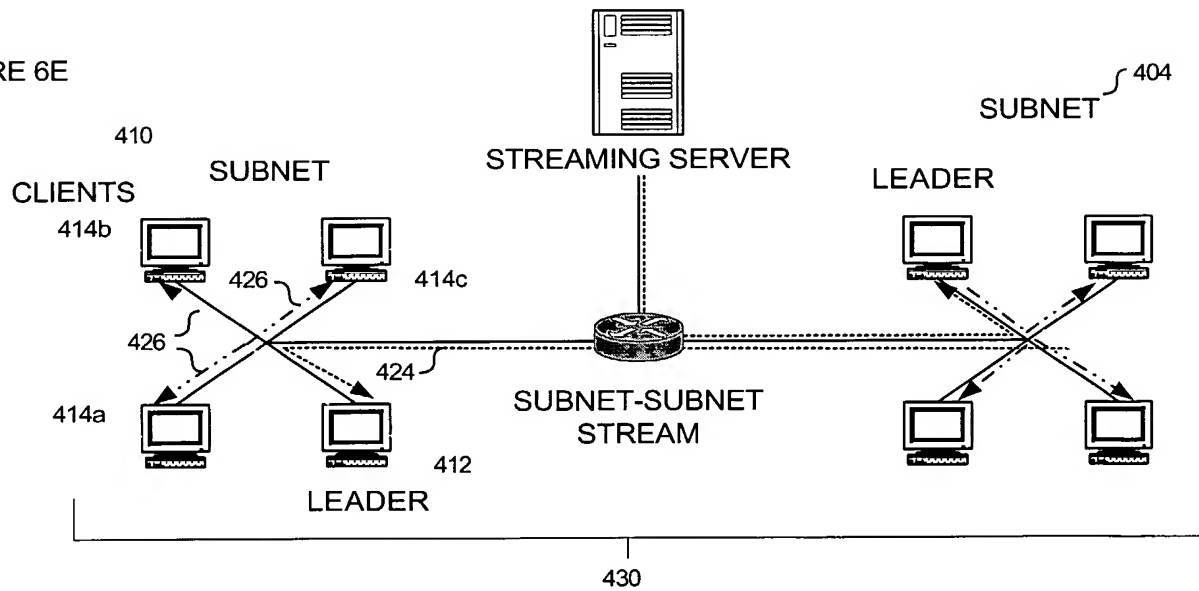


FIGURE 7A

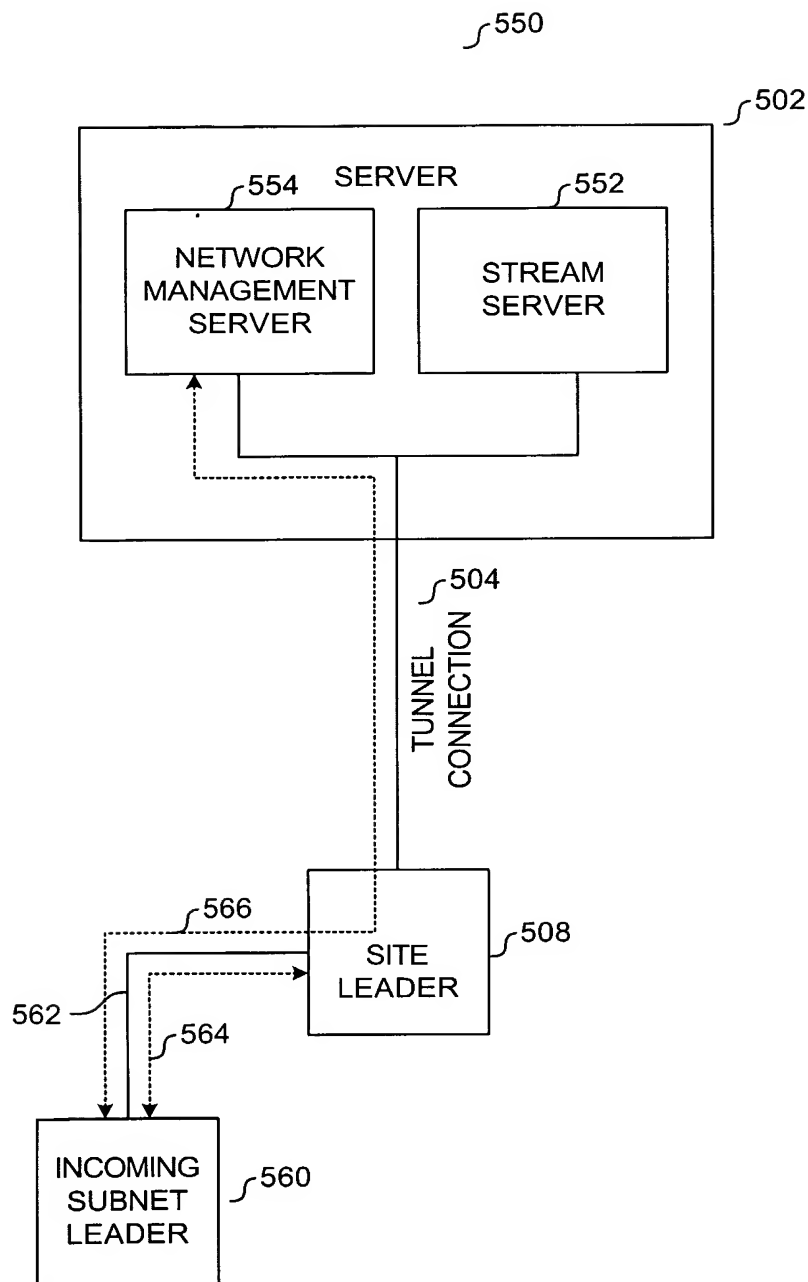


FIGURE 7B

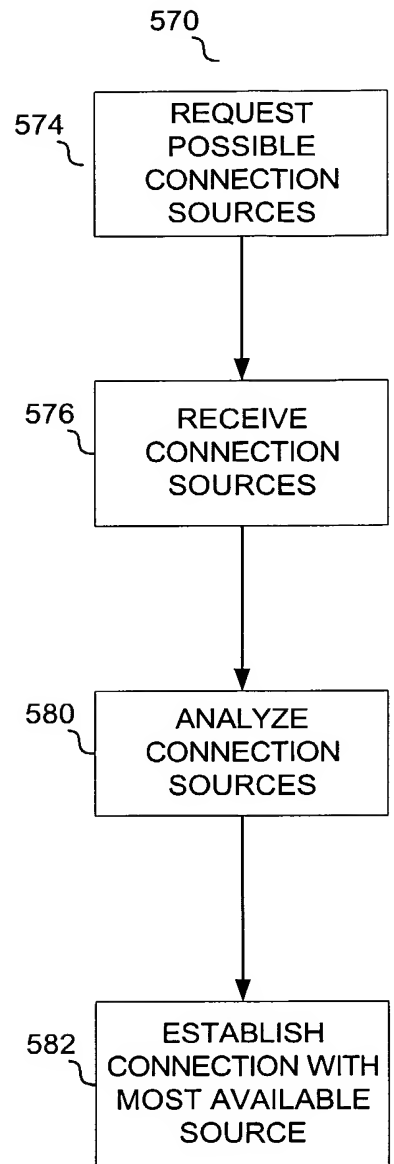


FIGURE 8A

FAN-OUT LIMIT ORGANIZATION

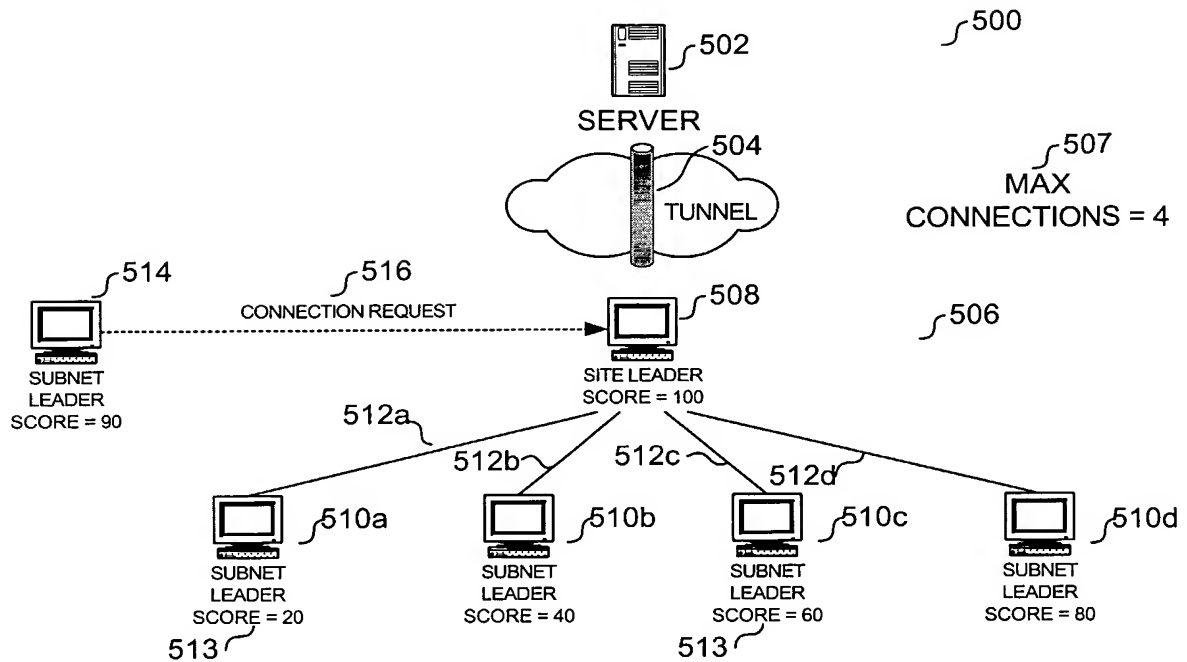


FIGURE 8B

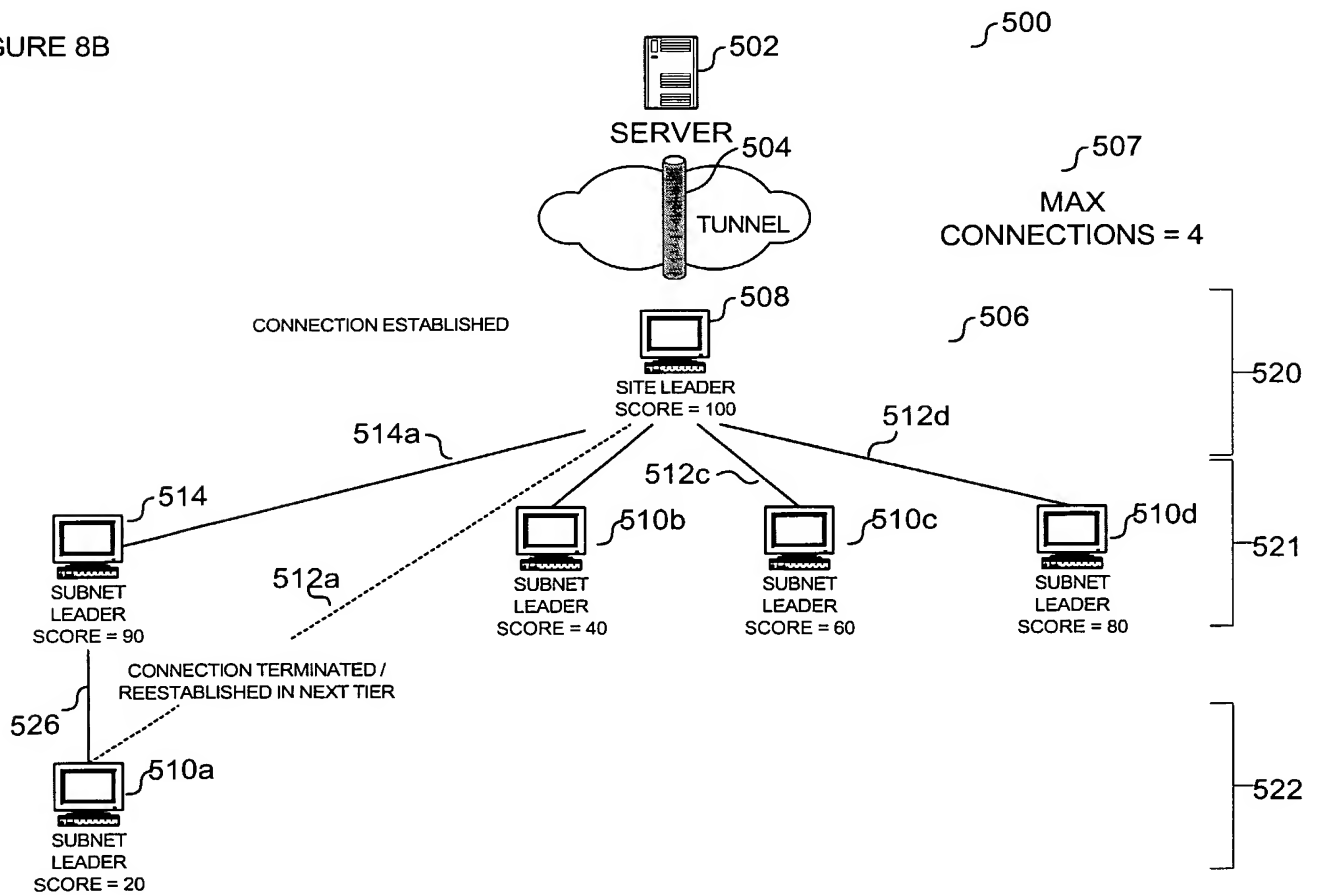


FIGURE 9A

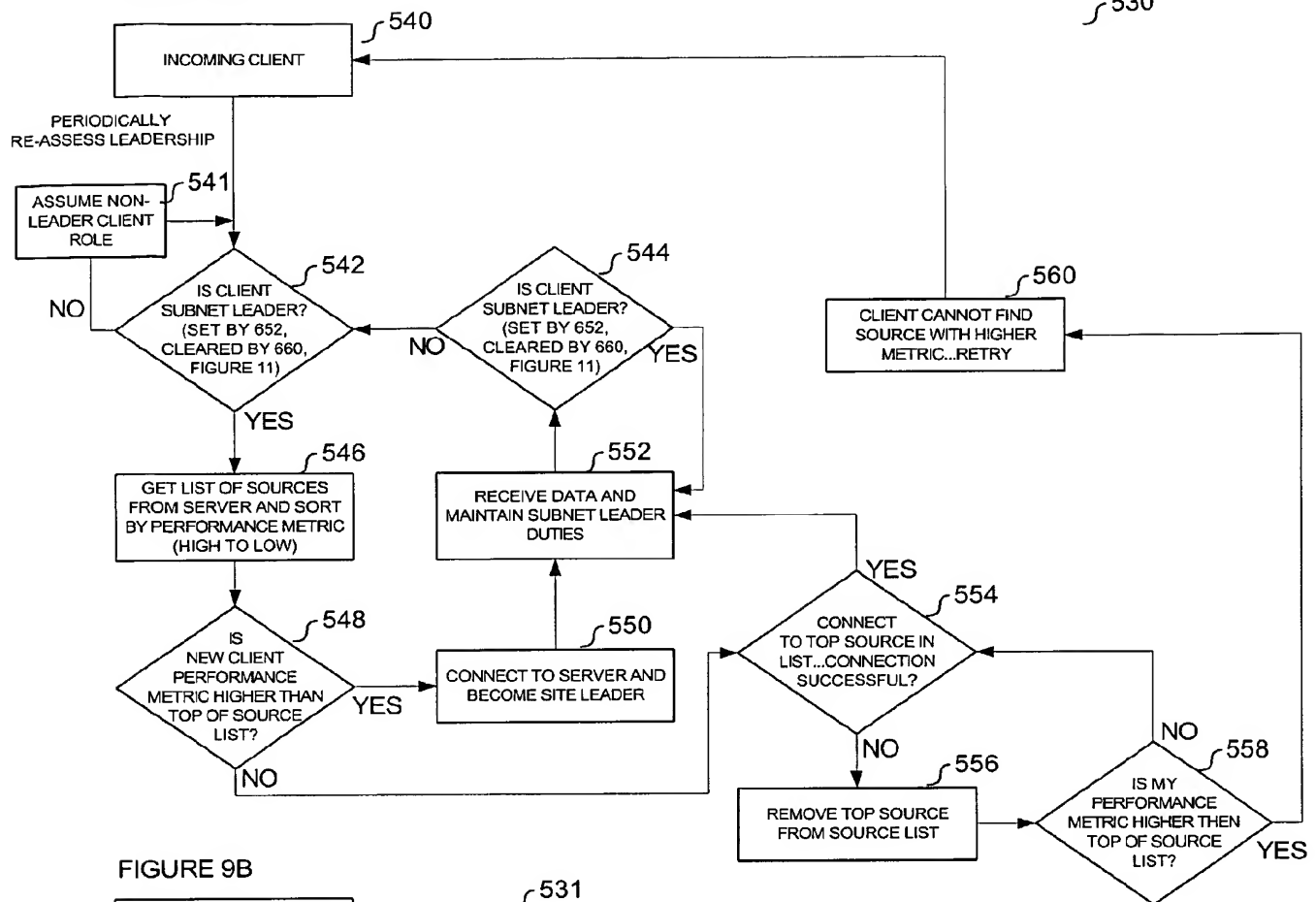


FIGURE 9B

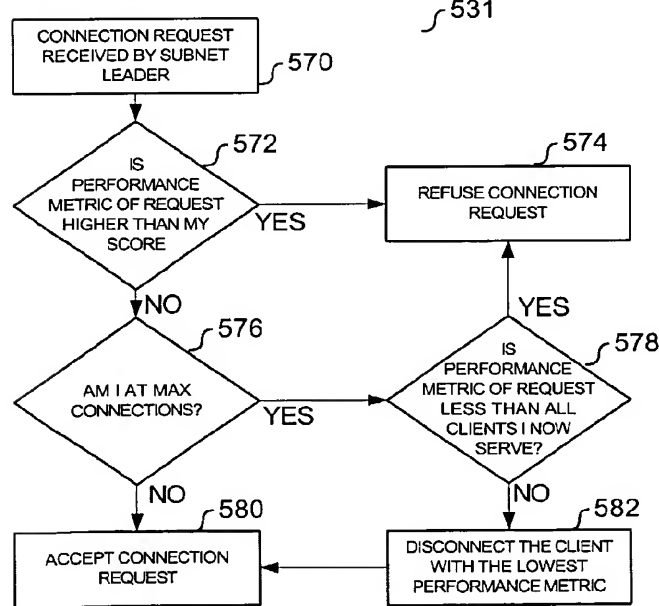


FIGURE 9C

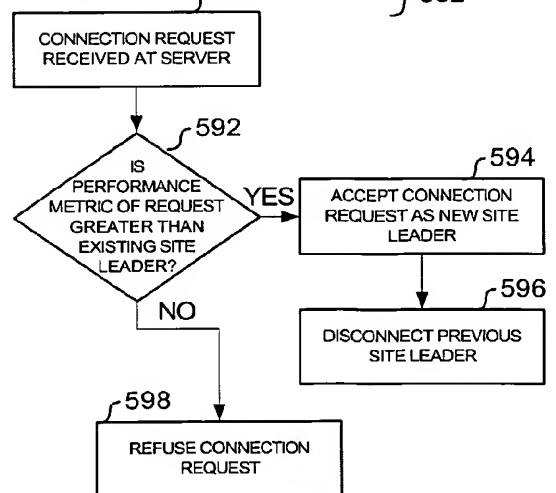


FIGURE 10

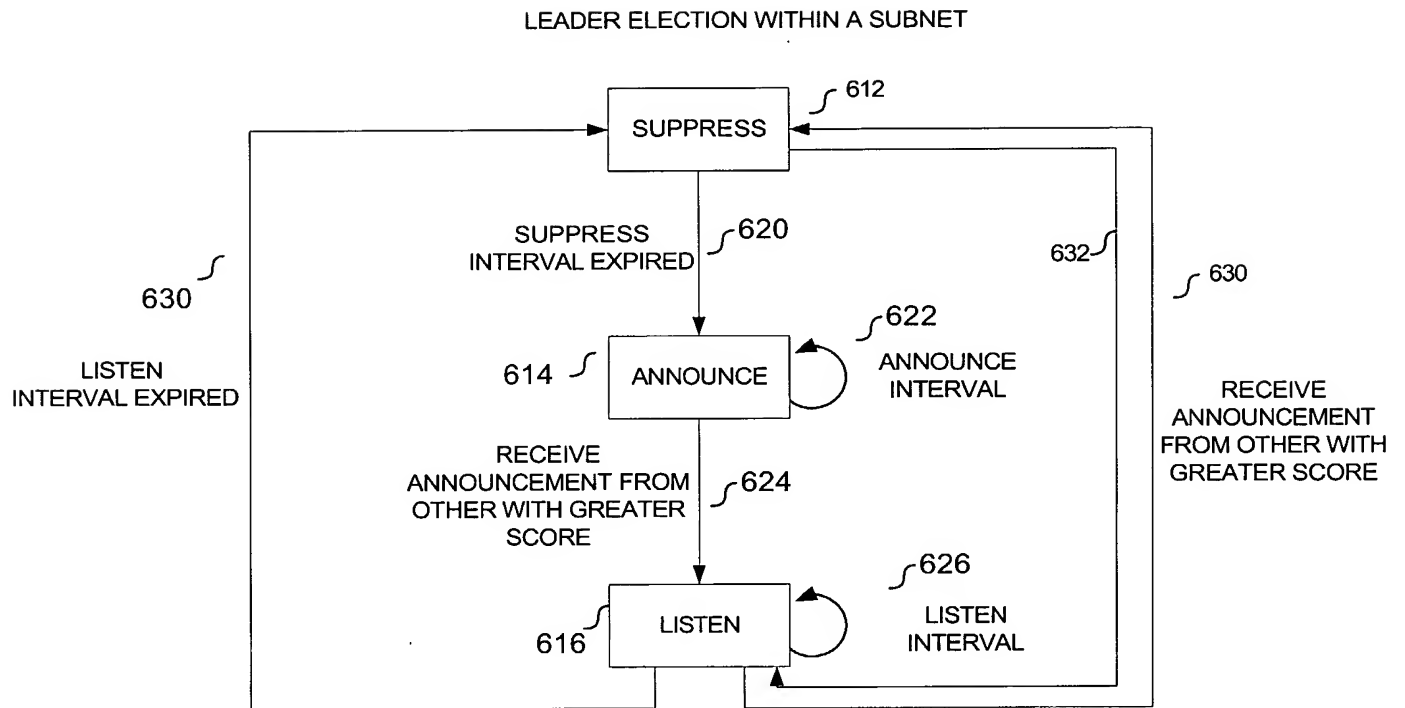


FIGURE 11A

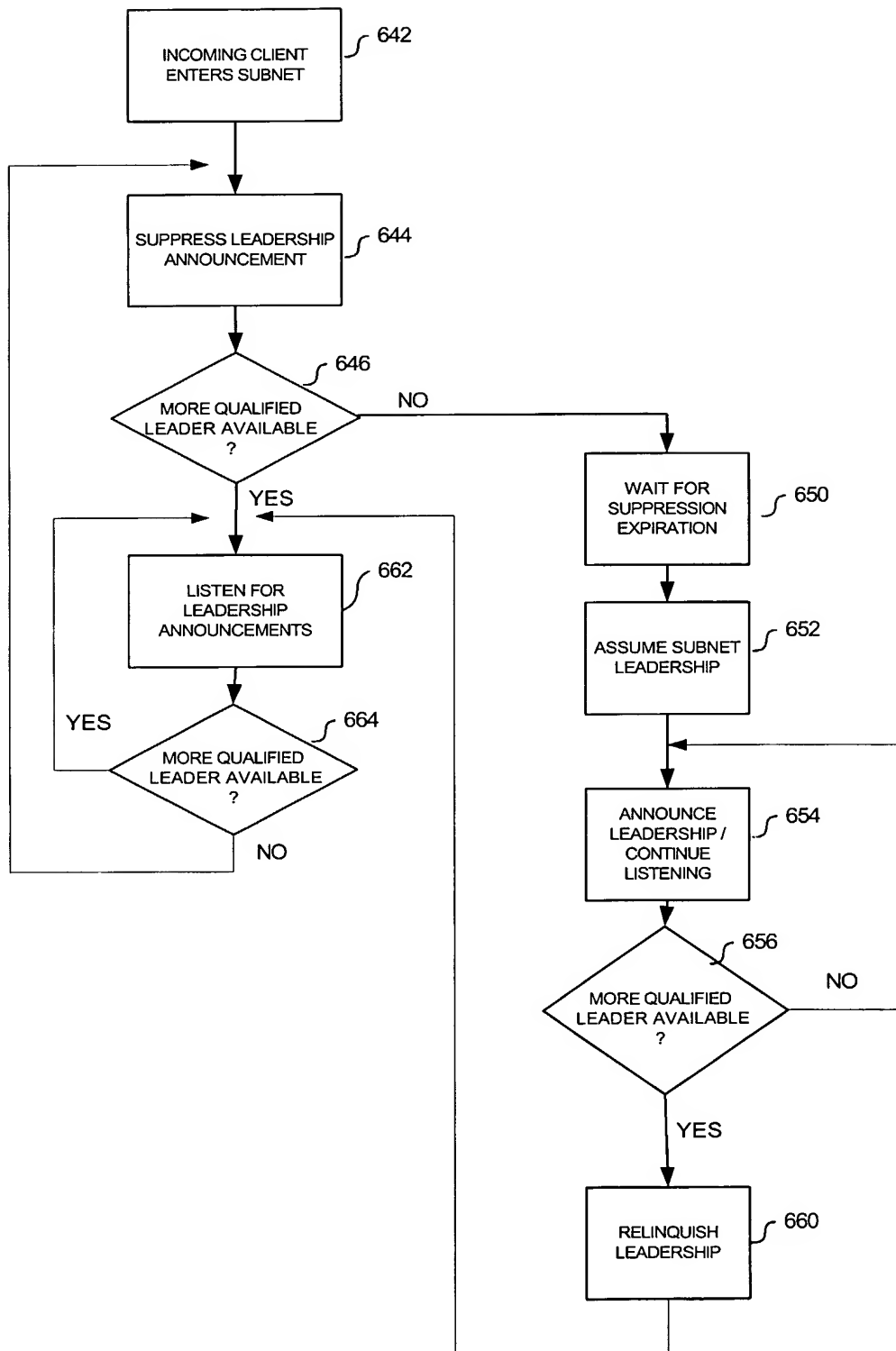


FIGURE 11B

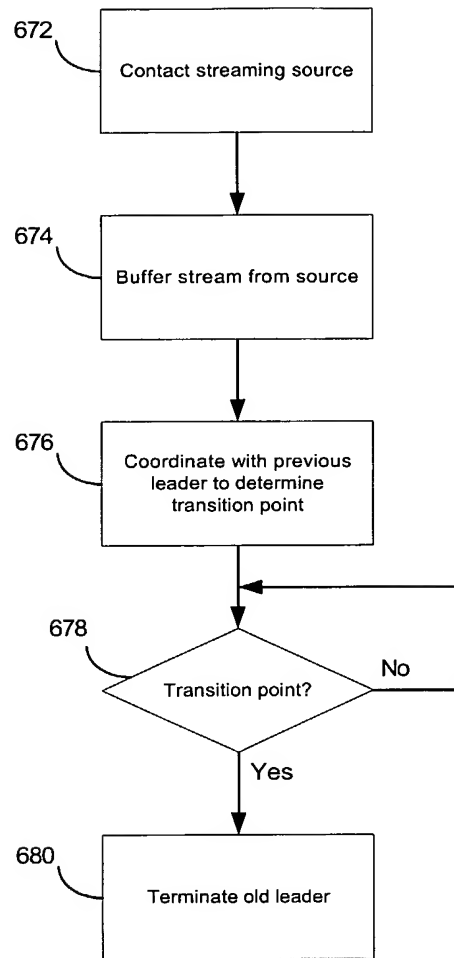


FIGURE 12A

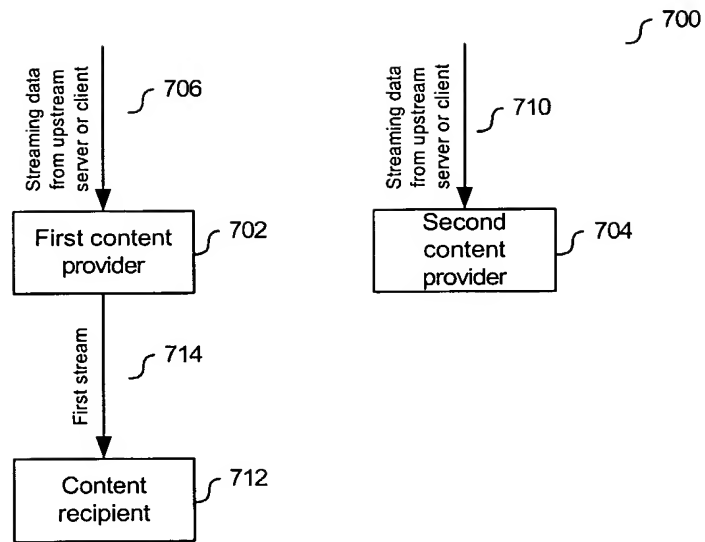


FIGURE 12B

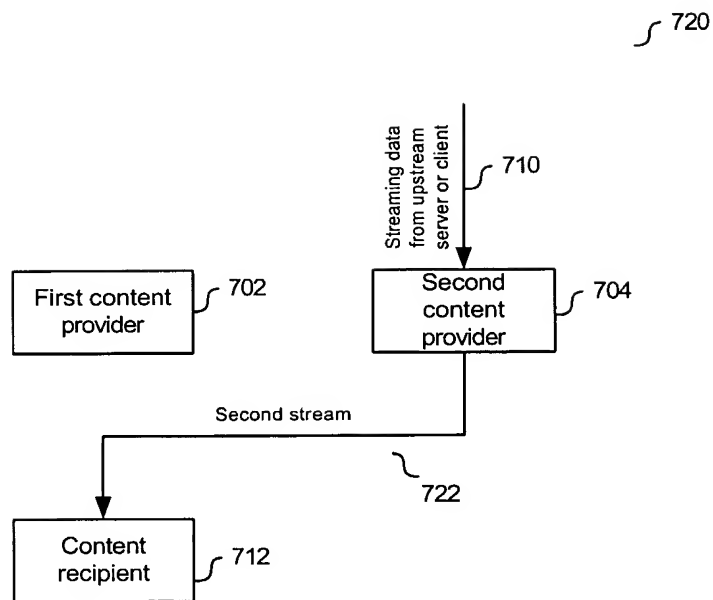


FIGURE 13

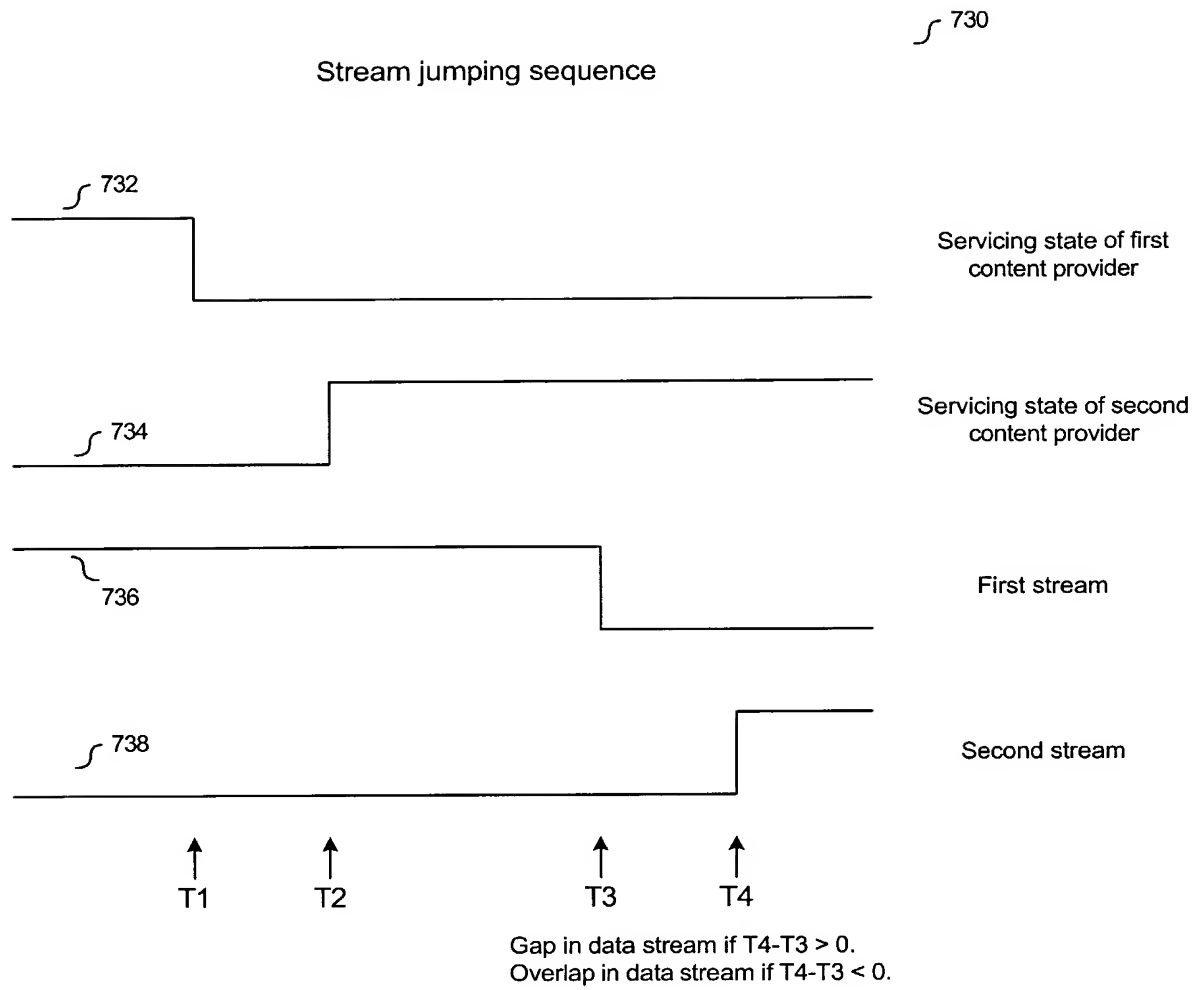


FIGURE 14A

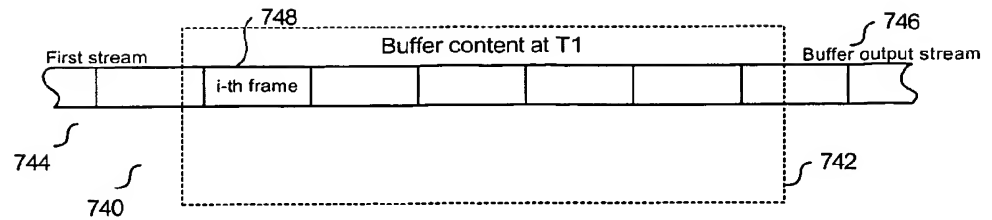


FIGURE 14B

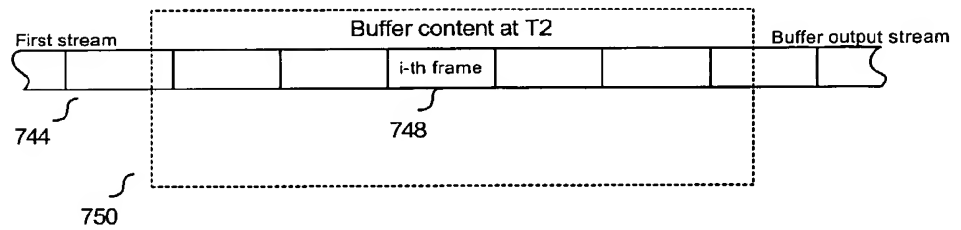


FIGURE 14C

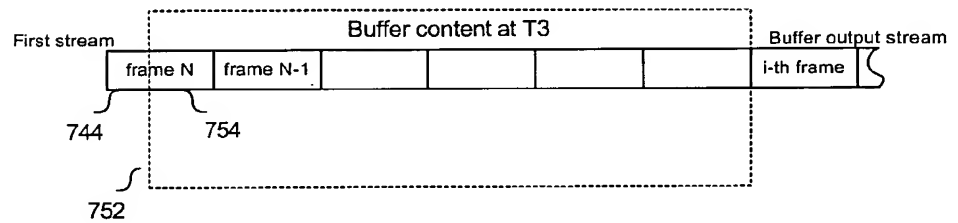


FIGURE 14D

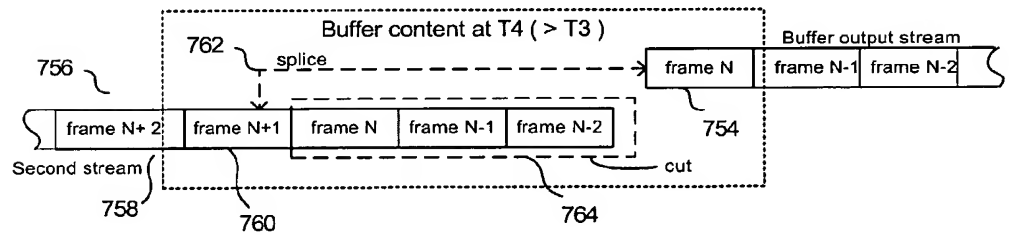


FIGURE 14E

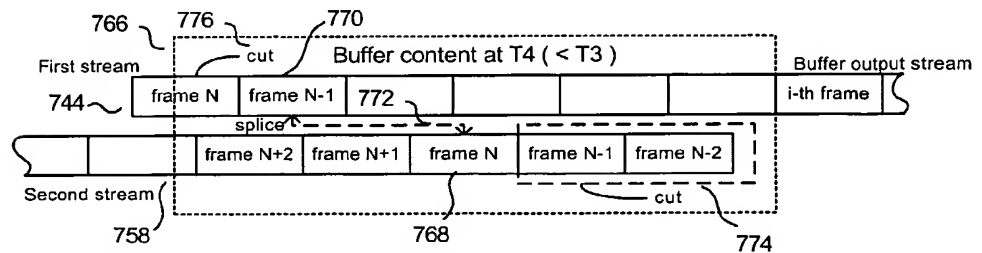


FIGURE 14F

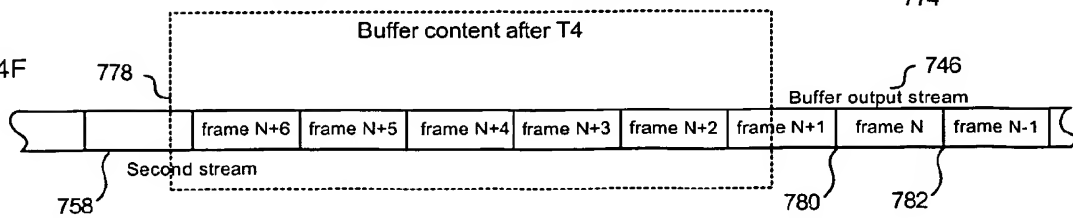


FIGURE 15A

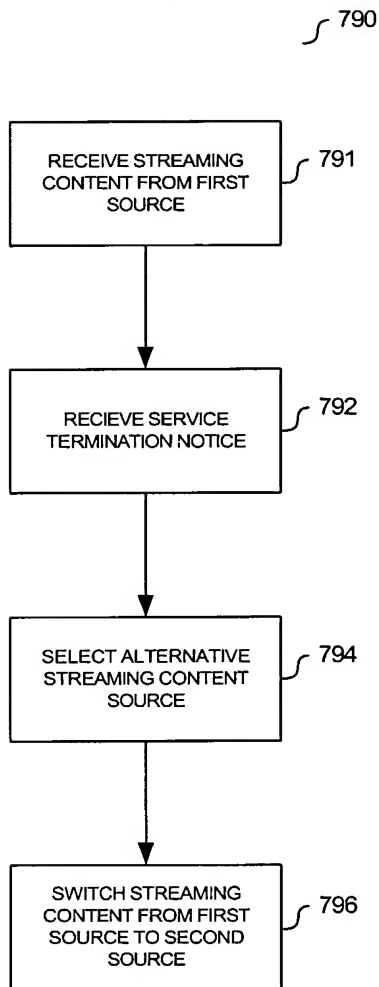


FIGURE 15B

